AIRCRAFT CONCEPTUAL DESIGN ENABLED BY A SET-BASED APPROACH FOR THE EXPLORATION AND BOUNDING OF NON-HYPERCUBIC DESIGN SPACES

A Dissertation
Presented to
The Academic Faculty

by

Justin R. Kizer

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
August 2016

AIRCRAFT CONCEPTUAL DESIGN ENABLED BY A SET-BASED APPROACH FOR THE EXPLORATION AND BOUNDING OF NON-HYPERCUBIC DESIGN SPACES

Approved by:

Prof. Dimitri N. Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Prof. Daniel P. Schrage
School of Aerospace Engineering
Georgia Institute of Technology

Prof. Graeme J. Kennedy
School of Aerospace Engineering
Georgia Institute of Technology

Prof. Sebastian Pokutta
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Jeffrey S. Schutte
GE Aviation, USA
General Electric Company

Date Approved: 22 June 2016

To my family,

your love and support made this possible.

ACKNOWLEDGEMENTS

Throughout the Ph.D. process I have felt extremely fortunate that I have had the opportunity to "stand on the shoulders of giants" to not only work toward academic achievement but also grow as a person. In writing these acknowledgments I wish to sincerely thank these giants who have hoisted me up and kept me from falling throughout this arduous but rewarding journey.

I must first extend tremendous gratitude to my advisor Prof. Dimitri Mavris. Doc, thank you for believing in me, taking me in and letting me join your amazing team. Thank you for expanding my horizons both technically and professionally. I learned far more from you than I ever thought possible and I think it will still take me quite some time to fully absorb and appreciate all the wisdom you have shared. Thanks to your tutelage I feel ready and able for any challenges my career may have in store.

Next, I want to sincerely thank Dr. Fayette Collier and all the folks involved in the NASA ERA and ERA ITD programs. I owe you thanks not only for funding me throughout my tenure at ASDL but also for being perfect examples of why I became an engineer. Working with you was a pleasure and something I will not soon forget. I feel that the results of the ERA program will truly contribute to making the world a better place and I thank you for allowing me a small part in that endeavor.

To my committee, thank so much you for your time and effort. I have valued your advice and mentoring. Thank you for challenging me and pushing me to do my best and make a contribution to my field that I can proudly stand behind. Last but not least, thank you for sitting through my presentations and reading this book!

To the many friends and ASDL colleagues I have made over these six years in

grad school, thank you for being right there in the trenches with me. Thank you for sharing your experiences and your precious time. I have had way more fun in grad school than I ever thought possible and I owe it to all you wonderful people. Ryan and Mike, thank you so much for being the best thesis work/support group I could have ever asked for, I only hope I helped you as much as you did me!

To my parents, thank you for instilling in me a burning curiosity and a lifelong passion for learning and education. Thank you for your kindness, your generosity, your patience and your morals. Thank you for always being there when I needed you and helping to make me into the man I am today.

Finally, to my wife Heidi, thank you for being you. Throughout this process and our life together you have always been my strongest supporter. No matter what life has thrown at us, you have worked tirelessly and selflessly to make sure I am happy, healthy and loved. This would not have been possible without you and all your incredible help. Everyday I am thankful I have you to light up my life. I am so grateful for all the adventures we have had and so excited for all the new ones to come!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS OR ABBREVIATIONS

AAO All At Once.

AFC Active Flow Control.

ANN Artificial Neural Networks.

$ANOPP$ Aircraft Noise Prediction Program.

$ANOVA$ Analysis of Variance.

ASE Adaptive Sampling for Exploration.

ASR Adaptive Sampling for Refinement.

ATE Adaptive Trailing Edge.

AVL Athena Vortex Lattice.

BAS Bounded Adaptive Sampling.

BAU Business As Usual.

CBC CheckerBoard Coarse.

$CBC2$ CheckBoard Coarse 2-D.

CBF CheckerBoard Fine.

$CBF2$ Checkerboard Fine 2-D.

$CDFS$ Constraint Defined Feasible Set(s).

$CMPGEN$ Compressor Map Generator.

$CORQ$ Corollary to the Overarching Research Question.

CVE Cross Validation Error.

DOE Design of Experiments.

DSE Design Space Exploration.

$DSE - DSM$ Decision Support Methodology for Design Space Exploration.

DSG Design Space Guidance.

DT Delaunay Triangulation.

DV Design Variable(s).

EDS	Environmental Design Space.
ERA	Environmentally Responsible Aviation.
FAA	Federal Aviation Administration.
FDR	Feasible Design Ratio.
$FLOPS$	Flight Optimization System.
HC	Hypercubic Classification.
HEC	High Expense/Consequence.
HS	Hypersphere Removal Constraint.
$HS2$	Hypersphere Removal Constraint 2-D.
HWB	Hybrid Wing Body.
$ICAO$	International Civil Aviation Organization.
I/O	Input/Output.
LCL	Linear Constraint Large.
$LCL2$	Linear Constraint Large 2-D.
LCS	Linear Constraint Small.
$LCS2$	Linear Constraint Small 2-D.
L/D	Lift to Drag Ratio.
LHS	Latin Hypercube Sampling.
$LHSMM$	Latin Hypercube Sampling Minimax Optimized.
$LHSRC$	Latin Hypercube with Reduced Correlation.
LPD	Levels Per Dimension.
LTA	Large Twin Aisle.
MET	Method Execution Time.
MI	Mutual Information.
MID	Mutual Information Delta.
$MILCA$	Mutual Information Least-dependent Component Analysis.
$NASA$	National Aeronautics and Space Administration.

n/d	Cases (n) / Dimensions (d).
NFN	Number of False Negatives.
NFP	Number of False Positives.
NHC	Non-Hypercubic.
$NLCL$	Non-Linear Constraint Large.
$NLCL2$	Non-Linear Constraint Large 2-D.
$NLCS$	Non-Linear Constraint Small.
$NLCS2$	Non-Linear Constraint Small 2-D.
$NPSS$	Numerical Propulsion System Simulation.
OML	Outer Mold Line.
ORO	Overarching Research Objective.
ORQ	Overarching Research Question.
$P3T3$	Pressure and Temperature Correlations.
PG	Prandtl-Glauert.
PMC	Pseudo-Monte Carlo.
RBF	Radial Basis Function.
RF	Random Forests.
RHM	Reduced Hypercube Multiple.
$RHM1$	Reduced Hypercube Multiple 1-D.
RHS	Reduced Hypercube Single.
$RHS1$	Reduced Hypercube Single 1-D.
RNG	Aircraft Range.
$RRFP$	Random Removal Fixed Percentage.
$RRND$	Random Removal n/d.
RSM	Response Surface Methodology.
SBD	Set-Based Design.
SCP	Successful Case Percentage.

$SeBBAS$ Set-Based Bounded Adaptive Sampling.

$SSMAO$ Sobol Sequence with Matousek-Affine-Owen scrambling.

SVM Support Vector Machine.

TPF Tolerable Percentage of Global Failures.

$TSFC$ Thrust Specific Fuel Consumption.

$VCTE$ Variable Camber Trailing Edge.

$WATE$ Weight Analysis of Turbine Engines.

ZFW Zero Fuel Weight.

SUMMARY

This thesis work developed a Design Space Exploration Decision Support Methodology $DSE - DSM$ applicable to arbitrary design spaces and ultimately utilized to enable the efficient conceptual design of advanced aircraft concepts. It was observed for design problems of interest that the resultant feasible design space could be non-hypercubic due to the potential presence of correlated variables and explicit and/or embedded constraints. Thus, to conserve computational effort in exploring and understanding this space, a methodology was proposed to first provide hypercubic classification for a given design space and then if necessary, efficiently bound this space, and ultimately provide guidance on where the feasible regions of the design space exist.

In order to classify a given design space, the Mutual Information metric was utilized to detect the presence of any non-hypercubic features within the design space which would manifest as correlations between design variables. Using a globally classified initial sample of the design space, Mutual Information was demonstrated through experimentation to be a reliable hypercubic classifier when provided sufficient design space resolution to resolve features within the space. In order to quantify this sampling resolution requirement, a new similarity parameter coined equivalent 'Levels Per Dimension' LPD was defined to account for the exponential nature of design space volume growth with the addition of dimensions.

Once classified, the characteristics of the given design space as well as the experimental apparatus and available resources were utilized to provide informed design space exploration guidance. Based on a combination of three metrics encompassing

the percentage of feasible designs within the initial sample, the result of the hypercubic classification and the relative expense and consequence associated with design space exploration, suggestions are provided on how to most efficiently and effectively continue to sample the given design space.

For scenarios in which the feasible space was determined to be non-hypercubic and of either significant expense or consequence to explore, the DSE-DSM methodology advocates the use of the Set-Based Bounded Adaptive Sampling, $SeBBAS$, Method. Leveraging set-based design principles, the SeBBAS method constructs Constraint Defined Feasible Sets, $CDFS$, with relevant variable subsets of the design space. Machine Learning classifiers are then employed on these CDFSs to bound the feasible regions with respect to each unique constraint. By constructing these boundings in only the dimensions relevant to the respective CDFS, the SeBBAS approach artificially increases available sample resolution and was shown to produce superior results for the bounding of and adaptive sampling of d-dimensional feasible design spaces compared to a strictly global approach.

Ultimately, the utility of DSE-DSM and associated SeBBAS approach were demonstrated for the conceptual design of a Large Twin Aisle Hybrid Wing Body aircraft within the Environmental Design Space modeling and simulation environment. Given a design problem in 50 dimensions in a final experiment, the DSE-DSM methodology was able to increase the percentage of feasible designs from 72.0 to 93.8 percent when compared to 'business as usual' Pseudo-Monte Carlo sampling after only a single iteration. Additionally, the methodology was able to identify and rank variables relevant to the non-hypercubic features present within the design space all without significant additional computational expense compared to 'business as usual'. Ultimately, this thesis through the use of DSE-DSM and SeBBAS demonstrated the capacity for a more timely and resource conservative approach for the conceptual design of advanced aircraft concepts as well as other problems with similar characteristics.

CHAPTER I

MOTIVATION

1.1 Challenges Facing The General Experimental Design Problem

In science and engineering, experiments are performed and design problems solved as a means to test hypotheses and discover relationships between inputs and responses. In both of these examples, experimental parameters, or design variables DV, are perturbed in an intelligent fashion in order to produce a set of results and then draw conclusions. Any experiment or design problem can thus be described as the process of varying some set of variables X, within some limits L, to produce a set of results R from which conclusions can be drawn. In general, the experimenter seeks to maximize the size of R in order to generate a more complete understanding of the phenomena being observed and add statistical significance to conclusions drawn using the data. However, the varying of experimental parameters manifests itself as a cost to the experimenter, both due to the size of X and the number of unique variable combinations to be considered (n). Therefore, a resource efficient way of exploring this experimental space is naturally sought to minimize this cost.

At present, statistical methods such as Design of Experiments DOE are utilized to intelligently distribute these unique variable combinations (designs) throughout the experimental design space such that for a fixed experimental budget each design will theoretically yield a high rate of return toward the understanding of the phenomena/function/behavior being observed with the experimental apparatus. These methods are powerful but derive much of their effectiveness from underlying assumptions made about the experimental design space. Chief among these empowering

assumptions, and common across many DOE types, is that the experimental design space can be generalized to a d-dimensional hypercube. That is, any design X^* to be evaluated exists and is feasible within the continuous space bounded by the limits $L = [L_{lower}, L_{upper}] \subset \mathbb{R}$ such that $X_i^* \geq L_{lower_i}$ and $X_i^* \leq L_{upper_i}$ for all $i = 1, 2, \ldots, d =$ number of design variables. Furthermore, the set of variables which comprise X are all assumed independent and thus are mutually orthogonal dimensions of the hypervolume which bounds the experimental design space. These assumptions along with the normalization/scaling of each dimension allow the translation of any experimental design space to a hypercube of dimension d. This general structure then allows for a DOE to be constructed that is agnostic to problem characteristics yet maintains geometric similarity and thus will not artificially bias any particular regions when linearly scaled (mapped) to the true experimental design space. These characteristics mean a specific DOE could be constructed and mapped to any number of experimental design spaces which share the same number of variables/degrees of freedom.

However, there are circumstances in which the enabling assumptions used to construct many DOE no longer hold. If the design variables chosen are not truly independent and exhibit any significant form of correlation then regions of the design space may be inaccessible. Furthermore, constraints based on the physics of the experiment/problem being considered may result in areas of infeasibility. Additionally, numerical constraints associated with computer experiments, such as convergence failures or numerical singularities, may further violate these assumptions and prevent areas of the experimental design space from being evaluated. These conditions and others of the like serve to transform the feasible experimental design space into what the author terms a 'Non-Hypercubic NHC design space', that is an experimental design space which can not be linearly scaled to a d-dimensional hypercube without losing information in the process.

As the typical modus operandi for sampling experimental design spaces efficiently relies on the construction of DOE based upon a hypercubic assumption, a natural question that arises is what are the consequences associated with treating a space as hypercubic and sampling it with a hypercubic design when it is in reality non-hypercubic? The first and most obvious consequence is the loss of some portion of the experimental budget due to design cases which are deemed infeasible or simply do not yield a usable response. Depending on the relative expense of the evaluation of each design this in itself may be cause for concern. In addition, even if such a loss can be tolerated resource wise, when propagated to the end goal of the design space exploration, serious issues may arise. The NHC space may feature variables which are unknowingly correlated or related through constraints that restrict the feasible design space. If regressions/surrogate models are being constructed utilizing the design space exploration as a training set, then models fit for this space will extrapolate over the regions made infeasible by NHC features. Because of these potentially significant consequences, there is the need for a design space exploration decision support methodology which acknowledges that an arbitrary design space need not be hypercubic. Such a methodology could provide guidance for design space exploration based on the characteristics of the problem and a more efficient path toward knowledge of the experimental design space.

This thesis seeks such a methodology that will provide decision support for the design space exploration for arbitrary design spaces and ultimately enable the efficient conceptual design of advanced aircraft concepts. Such a methodology will provide hypercubic classification and guidance on how to best sample the design space. Should the design space be NHC and require more advanced sampling techniques than provided by contemporary DOE, the methodology will utilize Set-based Bounded Adaptive Sampling to further explore and understand the design space. Ultimately the goal of this thesis is to enhance the efficiency of design space exploration for arbitrary

spaces and the understanding which comes as a result.

1.2 Growth and Challenges of Civil Air Transportation

The particular application motivating this investigation stems from the chalenges presented by the growing demand and projected growth of civil air transportation. This growth has begun to impose new requirements and reshape objectives involved in the design of future aircraft systems. The airliners of today and the near future differ from their predecessors in many ways, and through these differences and technological advances common motivational themes emerge. The civil aviation industry has seen the turbojet transformed into a turbofan and transition from low-bypass to high-bypass turbofan with projected technologies reaching for the ultra-bypass classification. Wing aspect ratios have grown from 7.1 on the Boeing 707 [56] to 11 featured on the 787 [42].

Historically these advances have been driven by performance with aircraft striving for superior range and payload capacity. More recently however, new legislative requirements and rising fuel costs have changed the design objective function to instead emphasize affordability and the minimization of environmental impacts. Such goals can be readily seen in meeting transcriptions from the International Civil Aviation Organization $ICAO$ and in stated objectives for future aircraft systems produced by the Federal Aviation Administration FAA and the National Aeronautics and Space Administration $NASA$ (see 1) [1, 2, 33]. Key to many of these goals is a reduction in aircraft fuel burn. Not only is burning less fuel for a given route economically advantageous for aircraft operators, but CO_2 and NO_X emissions scale proportionally with fuel burn.

Table 1: NASA Subsonic Transport System Level Metrics and Goals [33]

Technology Benefits	Technology Generations (Technology Readiness Level = 4-6)		
	N+1 (2015)	N+2 (2020)	N+3 (2025)
Noise (cum margin rel. to Stage 4)	-32 dB	-42 dB	-52 dB
LTO NOx Emissions (rel. to CAEP 6)	-60%	-75%	-80%
Cruise NOx Emissions (rel. to 2005 best in class)	-55%	-70%	-80%
Aircraft Fuel/Energy Consumption (rel. to 2005 best in class)	-33%	-50%	-60%

1.2.1 Setting a Goal: Fuel Burn Reduction

Burning less fuel is clearly desirable from both economic and environmental perspectives, thus the natural question arises, how can a reduction in aircraft fuel consumption be achieved? To provide insight on how to answer this question the Breguet Range Equation (Eqn. 1) can be examined as a canonical example [84].

$$RNG = \frac{V}{TSFC} * \frac{L}{D} * ln\left(\frac{W_{TO}}{W_{FINAL}}\right) \tag{1}$$

This is the standard form of the Breguet Range Equation for a jet aircraft in which performance, given by range RNG, is expressed as a function of vehicle system characteristics. The thrust specific fuel consumption $TSFC$ provides a representation of engine performance, the lift to drag ratio L/D represents aerodynamic performance of the vehicle and finally the weight fraction seen in the last term incorporates the structural and weight characteristics of the system. In this one powerful equation, many of the major objectives and trades of the field of aeronautical engineering can be seen. This equation, when rearranged, can also provide a clear picture of the options an aircraft designer may have when attempting to achieve a reduction in aircraft fuel burn. To make this apparent, constituent components of the weight terms must first

be examined as can be seen in equations 2 and 3.

$$W_{TO} = W_{EMPTY} + W_{PAYLOAD} + W_{FUEL} \tag{2}$$

$$W_{FINAL} = W_{TO} - W_{FUEL} = W_{EMPTY} + W_{PAYLOAD} \tag{3}$$

In these equations the takeoff weight (W_{TO}) of the aircraft is fully described by three weight groups. W_{EMPTY} represents the empty weight of the aircraft (including any trapped fuel and necessary fluids) without any mission fuel or payload. This term is almost entirely fixed and determined by the weight of the airframe structure, engines and furnishings within the aircraft. $W_{PAYLOAD}$ refers to the mission payload, and for a civil transport includes all passengers, crew members and luggage. Finally, W_{FUEL} represents the weight of the fuel to be consumed during the mission (neglecting any reserves which can be book-kept in $W_{PAYLOAD}$). Thus, after the mission has been flown and the aircraft landed, the final weight (W_{FINAL}) of the vehicle can be described as the takeoff weight less the mission fuel or alternatively the sum of the empty weight and mission payload. Rearranging equation 1 (see Eqn. 4) and then substituting in these weight expressions for the terms in the weight fraction will ultimately yield an expression for the weight of the mission fuel expressed as a function of the aircraft characteristics (Eqn. 5).

$$\frac{W_{TO}}{W_{FINAL}} = exp\left(RNG * \frac{TSFC}{V} * \frac{D}{L}\right) \tag{4}$$

$$W_{FUEL} = (W_{EMPTY} + W_{PAYLOAD}) * \left[exp\left(RNG * \frac{TSFC}{V} * \frac{D}{L}\right) - 1\right] \tag{5}$$

1.3 Achieving a Fuel Burn Reduction

In equation 5 the influence of the aircraft characteristics on the required mission fuel can be clearly seen. Recalling the objective to minimize the fuel burned by the aircraft for a given mission (fixed range and mission payload), it can be observed that this can be accomplished through the minimization of the empty weight and thrust specific fuel consumption and the maximization of the lift-to-drag ratio. This observation is very powerful as it provides the aircraft designer three different terms with which to operate on and improve in the specific directions to yield a desired reduction in mission fuel burn. The next question that then naturally arises is: how does the aircraft designer manipulate these three terms representative of vehicle characteristics?

To answer this question, three primary avenues exist for vehicle performance improvement:

1. Vehicle Optimization
2. Operations Optimization
3. Technology Infusion

The first avenue of vehicle optimization encompasses activities which seek to optimize existing aspects of the aircraft system through design choices. These activities are largely performed before the aircraft has been built and often draw upon lessons learned from earlier vehicles as well as emerging research and development. Vehicle optimization often relies upon heavy use of modeling and simulation as the design space is scoured for optima. Examples of vehicle optimization in action are utilizing aerodynamic optimization for vehicle outer mold line OML and wing shape optimization and employing structural optimization for primary wing structure design. The second avenue, operations optimization, includes activities which seek the optimal

procedure in which the aircraft carries out its mission. This could include anything from minimizing taxi time through improved airport queuing to mission trajectory optimization. Determining the optimal cruise climb schedule for an aircraft within air traffic control requirements is an example of operations optimization. Lastly, vehicle performance can be improved with the introduction of new technologies. While the previous two approaches are typically more evolutionary in nature, with largely incremental improvements occurring over time, the introduction of a new technology to an aircraft system can provide revolutionary jumps in performance. This avenue can often show great promise, but such benefits often come at the cost of increased uncertainty and risk [59]. To mitigate this risk, technology infusion also relies heavily on modeling and simulation to propagate technology level impacts to system level performance [58, 60, 59].

1.3.1 Technology Development and Infusion: NASA ERA

There are many examples of technology development programs actively using this third avenue to seek aircraft system performance improvement. Some examples of this include the FAA Continuous Lower Energy, Emissions and Noise (CLEEN) program, initiatives undertaken by large scale integrators (such as Boeing, UTC, Lockheed), studies performed by national research institutes such as South West Research Institute (SWRI), and the NASA Environmentally Responsible Aviation ERA program [104, 81]. Many of these programs are exploring immature technologies that have the promise of a reduction in aircraft fuel burn. The Boeing ecoDemonstrator program for example has examined an adaptive trailing edge ATE and a vertical tail augmented with active flow control AFC among many other technologies for the very purpose of improving aircraft performance and thus reducing fuel burn [70, 71]. The NASA ERA program is charged with evaluating a large portfolio of emerging technologies set to be integrated into the fleet within the N+2 time frame [104].

In order to evaluate the potential impact and system level effects of such a large set of immature technologies as being considered within NASA ERA, it is necessary to rely upon physics-based modeling and simulation to estimate performance of future vehicles. Such a task requires the incorporation of advanced models which can capture the physics affected by such technologies and realistically simulate their effect on a sized aircraft. Furthermore, with such and expansive set of technologies and airframe concepts considered, a huge design space with variables mapping to both aircraft parameters and technology metrics must be explored. These items highlight the complexity of such a design problem, yet they are not unique to this particular example. Through a functional decomposition of the required elements of this problem, its characteristics can be listed and requirements can be described for solving a problem of this class. This approach can provide guidance for potential pathways which may be applicable for the conceptual design of an advanced aircraft concept which can meet the aggressive requirements of the future.

1.4 Generalizing the Problem

While the physics-based conceptual design and analysis of an advanced civil transport aircraft concept is a specific problem, it shares many characteristics with other complex problems. By examining these characteristics, this unique problem can be categorized among other like problems that utilize established techniques to aid in their exploration and ultimate solution. The aircraft design problem was found to have the following general attributes:

- Coupled multidisciplinary problem
- Computational analysis required
- Moderate fidelity analysis required to capture relevant physics
- High dimension (many design variables)

- Numerous constraints
- Multiple constraint types

  known and unknown

  physical and non-physical

  linear and non-linear
- Potentially correlated design variables
- Subject to multiple repeat analyses as assumptions and models are revised

1.4.1 Categorization: Computationally Expensive

As it is prohibitive to explore this problem with physical experimentation, and historical data does not exist for advanced concepts with emerging technologies, this problem needs to be explored with computational analysis. Furthermore, due to the coupled nature of the physical disciplines governing the design of an advanced aircraft concept, at least moderate fidelity methods should be utilized to ensure all relevant physics are captured. In addition, as is common with many design problems attempting to explore a large design space, the problem is inherently one of high dimensionality. Thus, to explore a new design space of high dimension, while requiring moderate fidelity methods, makes the proposed problem likely expensive in terms of the time expenditure of computational resource allocation.

1.4.2 Categorization: Non-Hypercubic Design Space

The nature of the problem also dictates that many constraints can be expected to subdivide the design space, and that the constraints themselves may be varied in nature. Certain constraints will be tied to aerodynamic requirements, while others focus on propulsive considerations, others still, will likely emerge due to the complicated interactions between the disciplines. Some constraints may not even be physical

and arise due to the nature of the computational models and assumptions used. As computational models are emulations of true physics, they are by nature inexact and thus not applicable for every regime or range and combination of variables considered. Additionally, many design variables to be considered because of their relevance to aircraft design may not be truly independent. This characteristic further changes the shape of the feasible design space as variable correlations lead to trends and biasing in the feasible design space. These constraints and variable correlations have the effect of distorting the feasible design space such that it is no longer a regular hypercube (example shown in Fig 1). That is, the feasible design space is not simply defined by upper and lower variable bounds in each of the d-dimensions, but rather dictated by a combination of these bounds, active constraints and variable inter-dependencies. In this instance, constructing a bounding of the design space which determines the boundaries of this irregular feasible space may assist in making the exploration of the problem more efficient.

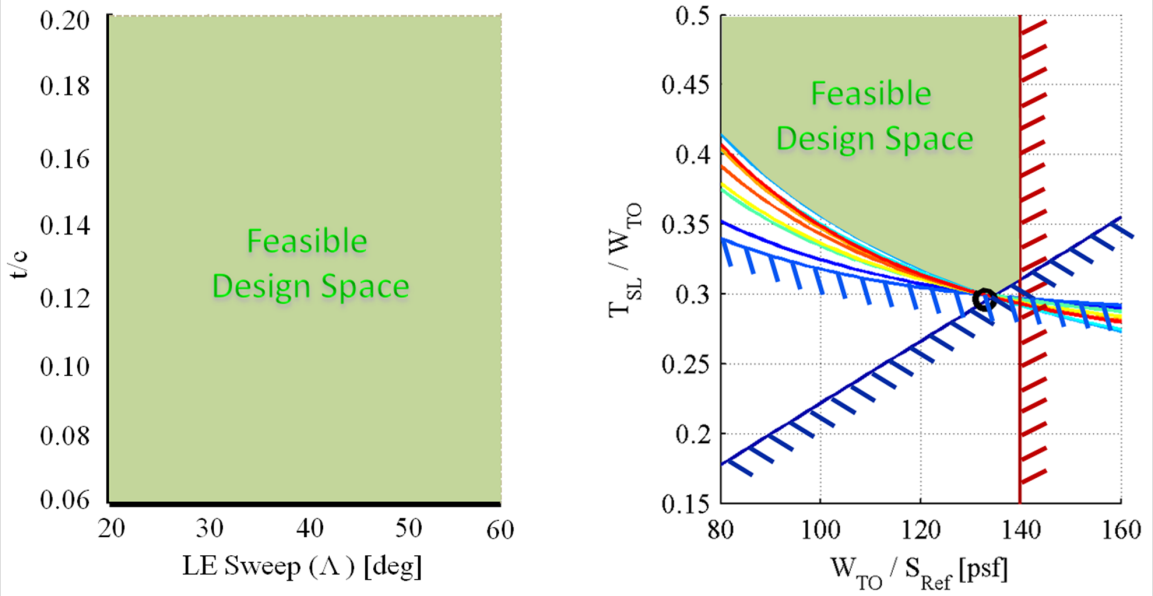

Figure 1: Regular Hypercubic Design Space vs. a Non-Hypercubic Space Defined by Numerous Non-linear Constraints from a 2-D Energy Based Constraint Analysis [48, 62]

1.4.3 Categorization: Repeated Exploration

As the design of advanced aircraft concepts often model technology still in developmental phases, many assumptions must be made about performance and requirements. Technology applications and performance impacts will be updated through the efforts of technology development programs, and based on this new data, the output of computational models representing its behavior will evolve as well. In response to these updates, feasible regions of the design space may shift and previous designs may lose their optimality. To address this fluidity, the design space will require revisiting as computational models and constraints are updated.

1.4.4 Problem Classification

All the aforementioned attributes can be used to provide a classification for this particular problem. The conceptual design problem of an advanced aircraft concept belongs in a class of problems which are **computationally expensive**, expected to have design spaces which are **non-hypercubic** and **repeatedly explored**. This classification leads to an Thesis Research Objective which drives the efforts of this thesis:

1.5 Thesis Research Objective

The objective of this thesis is to search for, and if necessary create, a methodology that will facilitate conceptual design and analysis of problems which are computationally expensive, non-hypercubic in nature and likely to be revisited multiple times. In order to accomplish this goal, this thesis will strive to illustrate that current methods are lacking in their ability to efficiently provide accurate representations of the feasible design space for problems of this class. Then, to bridge this gap, a methodology is proposed which will hopefully provide a greatly improved representation and understanding of the design space compared to those generated using existing methods

with similar computational resources. This superior design space representation is expected to allow for the production of higher quality regressions/surrogates, more efficient optimization and future Design Space Exploration DSE, and a greater understanding and capacity to visualize constraints, and relationships between variables and computational model limits. The methodology will then be applied and tested against competing current methods on the problem of interest: the design space exploration for the conceptual design of a Large Twin Aisle LTA Hybrid Wing Body HWB passenger transport aircraft. This thesis will then suggest that the methodology used to make this analysis realizable can be generalized to problems of a similar class and thus has wide applicability for expensive and complex problems.

1.5.1 Overarching Research Question and Corollary

With this objective in mind, an overarching research question can be formulated which highlights the critical investigations proposed in this thesis. This question will be revisited throughout this work to provide guidance and aid in the derivation of lower level research questions and hypotheses.

Overarching Research Question ORQ:

What methodology, consisting of what elements, should be used to provide decision support for the exploration of design spaces and solution of problems which are **computationally expensive**, **non-hypercubic** and must be **repeatedly explored**?

Corollary to the Overarching Research Question $CORQ$:

How should such a methodology be applied for the conceptual design of an advanced civil transport aircraft?

1.5.2 Statement of the Overarching Research Objective

To begin to answer these questions, the thesis research objective is reformulated in a more focused and succinct way to better guide the investigation.

> **Overarching Research Objective** *ORO*:
>
> This thesis seeks a general methodology which addresses the gaps in current practices for conceptual design and analysis of problems which are computationally expensive, non-hypercubic, and subject to re-visitation. This new methodology should be able to classify these designs spaces as non-hypercubic and then produce a **resource efficient** representation or **bounding of the feasible design space** such that the characteristics of design space can be more easily understood. This design space knowledge will enable surrogate generation, optimization, visualization and efficient future exploration of the design space.

In order to meet this research objective, motivating questions (which are subsets of the overarching research question and corollary) will be posed throughout this thesis to define certain aspects of the problem and determine what is needed to construct a methodology which enables its efficient solution. These questions will be directed at determining the appropriate elements of the methodology used to classify and ultimately bound the design space to enable the conceptual design and analysis of the problem of interest. When applicable, these questions will be answered through literature review, observation and inference. However, when literature or existing techniques prove insufficient to answer particular research questions, hypotheses must be generated and ultimately tested through experimentation to provide answers.

CHAPTER II

BACKGROUND

2.1 Understanding the Challenges of the Problem

As mentioned in the overarching research question, the general problem of conceptual design of advanced aircraft concepts was classified as **computationally expensive**, **non-hypercubic** and subject to **repeated exploration**. These characteristics were explored each in detail to understand their potential causes and solutions. Ultimately research question were posed which would drive experimentation to address these issues within the methodology being presented.

2.1.1 Computationally Expensive

The design problem of interest for this thesis work was deemed computationally expensive, but what makes this so? As posited in the previous section, the computation model/tool characteristics necessary to enable a multidisciplinary physics-based analysis which adequately captures the relevant information for each of the disciplines within the conceptual design phase is part of the equation, but model choice alone does not make a problem expensive. So to address this inquiry another motivating question can be formally posed:

> **Motivating Question 1 (MQ1)**:
>
> What makes the physics-based computational design and analysis of an advanced aircraft concept computationally expensive?

The analysis of an advanced aircraft design space is both multidisciplinary and highly combinatorial [93], and thus the determination of the feasible space is often a resource intensive task. Even at the conceptual stage, to generate a realistic design, the

integration of aerodynamics, propulsion and structural (largely weight distribution) analyses must be performed and in an iterative nature, such that a feasible vehicle may be converged upon. If an advanced concept (likely using emerging technologies) is to be considered, historical data cannot be relied upon to provide accurate estimates as surrogates for these analyses. Additionally, as physical testing is infeasible at this stage of design, due to the sheer number of design alternatives to be considered (specified by a unique combination of design variables), physics-based computational modeling is often employed. In order to explore the design space thoroughly and accommodate requirements and objective functions, it is often desired that the computational models employed be parametric in nature. A parametric model enables rapid and semi-autonomous design space exploration, however, even with the use of such a model, there is a limit to what may be practically examined in search of the feasible design space. This limit exists due to the non-negligible computational resources required to evaluate these integrated models in addition to the size of the design space being explored. To provide an appreciation for the dimensionality of this problem, a example list of a small subset of typical conceptual design variables utilized within such models is presented:

1. Wing Aspect Ratio
2. Wing Area
3. Induced Drag Factors
4. Subsonic Drag Factors
5. Wing Transition Reynolds Numbers
6. Thrust to Weight Ratio (prop)
7. Takeoff Thrust
8. Fan Pressure Ratio
9. Max Turbine Inlet Temperature
10. Low Pressure Compressor Pressure Ratio
11. High Pressure Compressor Pressure Ratio
12. Customer Bleed Air Required
13. Combustor Efficiency

14. Wing Weight Factors
15. Fuselage Weight Factors
16. Engine Weight Factors
17. Design Payload Weight
18. Design Altitude
19. Design Mach Number
20. Design Gross Weight

Examining this list and further realizing that many of the items could be described as summary parameters which may be decomposed in detail by any number of lower level design variables, it can be observed that this problem is quickly becoming one of high dimensionality, and with the large ranges for each of these variables to be considered, one which suffers from the 'curse of dimensionality' [13]. In some extreme cases, upwards of 50 to thousands of design variables may be present just for some design problems, particularly those involving structural design and/or aeroelastic effects [11, 100, 20, 37, 72]. In addition, each unique design must be iteratively analyzed (using all relevant physics-based modules) in the multidisciplinary aircraft design environment and converged. Based upon these conditions, a notional required computational expenditure to exhaustively explore a given design space can be estimated using the following equation:

$$\text{Computational Time Required} = d^{FL} * IPC * TPI \tag{6}$$

Where d is the number of design variables (or factors), FL is the number of levels or settings per factor, IPC is the average number of iterations required to converge the main design loop (which contains all the discipline based modules) and TPI is the average time per iteration. Putting numbers to these values can provide an estimate of the computational expenditure required to evaluate each unique design (i.e. conduct a brute force exploration of the design space). For d, an approximate value is 20 using the above list of design variables for reference (this is a conservative estimate for most practical design problems). In order to capture any non-linear

behavior in responses, at least three factor levels should be evaluated for each design variable, thus FL = 3. The number of average iterations for vehicle sizing convergence per case can be optimistically estimated at two. Lastly, based upon the author's experience utilizing a conceptual aircraft design computational environment *EDS*, a conservative estimate for the time per iteration can be approximated at 5 minutes on an individual workstation (Benchmark: Intel i7 processor, 8 GB RAM, Windows 10 64bit). Substituting these numbers into equation 6 yields the following estimate for computational expenditure expressed in CPU time:

$$\text{Computational Time Required } = 20^3 * 2 * 5 = 80,000 \text{ minutes} = 55.55 \text{ days} \quad (7)$$

While many assumptions were made to arrive at this figure, it can clearly be seen from this example that attempting to solve this problem in this particular fashion is appreciably computationally expensive, and likely prohibitively so unless one has the use of significant distributed computing resources at their disposal. Referring to equation 6 and observing the output when appropriate numerical substitutions are made (Eqn. 7) allows MQ1 to be answered:

Answer to MQ1:

The physics-based computational design and analysis of an advanced aircraft concept is made computationally expensive by a combination of factors. These factors include the high dimensionality of the design space, the design variable ranges considered, the iterations required for aircraft sizing convergence and the individual design case run time using state-of-the-art multidisciplinary, physics-based modeling and simulation environments for the conceptual design of advanced aircraft concepts.

With the factors identified that contribute the computational expense of the problem, and reasonable and representative assumptions made for the values of all the

variables in equation 6, it can be seen such a problem may likely be unmanageable if approached and sampled unintelligently. Brute force exploration of the design space is very likely prohibitive. Inefficient design space exploration which results in the loss or failure of an appreciable percentage of the total number of designs considered would constitute a intolerable waste of computational resources, particularly if regression/-surrogate models are to be constructed from this data. Ultimately, should features exist within the design space which deny or make infeasible significant regions, design space exploration for such a problem utilizing business as usual DOE techniques such as LHS or PMC sampling may likely be found lacking.

2.1.2 Non-Hypercubic Design Space

With the expense of the physics-based computational design and analysis of advanced aircraft concepts illustrated, the next challenge to be illuminated is the non-hypercubic nature of the feasible design space. First, it is important to reiterate what is meant by the term 'hypercubic' and ultimately a non-hypercubic design space. "A hypercube is the generalization of a 3-cube to n dimensions" [105]. It is a closed, bounded, convex geometry which encompasses the space between limits specified by parallel lines in each dimension which are perpendicular and of the same length to limits in all other dimensions. Figure 2 illustrates hypercubes in dimensions ranging from d = 1 to d = 3.

Mathematically the d-dimensional hypercube can be described as the bounded space which consists of the points:

$$\{(x_1, x_2, \ldots, x_d) \subset \mathbb{R}^{\mathrm{d}} : x_{iLower} \leq x_i \leq x_{iUpper}\} \tag{8}$$

Where for the unit hypercube: $[x_{iLower}, x_{iUpper}] = [0, 1]$.

When speaking of design spaces, hypercubes are a very useful construct. When

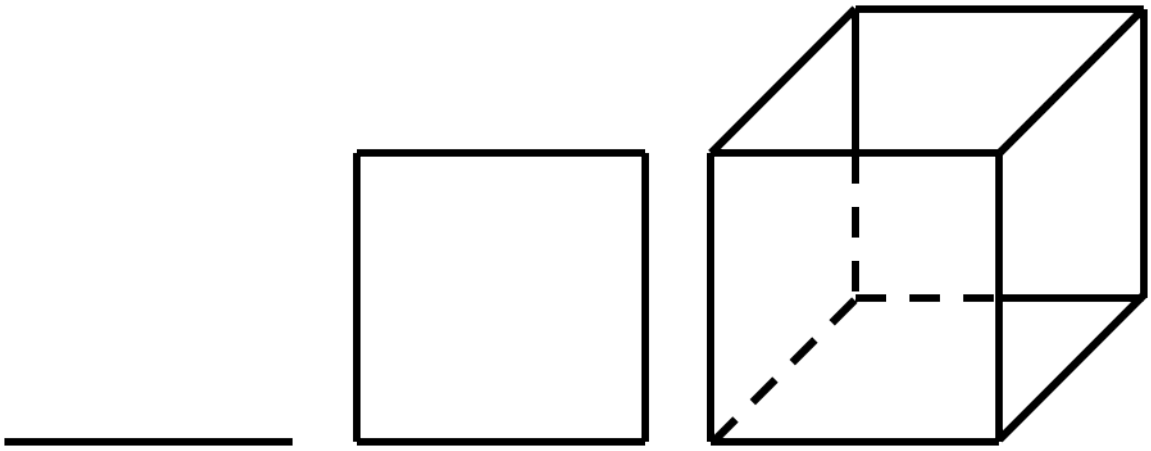

Figure 2: Examples of 1-D, 2-D and 3-D Hypecubes: Line, Square and 3-Cube Respectively

assembling a design space, design variables are assigned separate and mutually orthogonal dimensions. To bound the design problem, each of these design variables is given a range over which it may be varied, defined by lower and upper limits on its value($[x_{iLower}, x_{iUpper}]$). When the d-dimensional orthogonal basis created by the d design variables is bounded by each of the variables' respective limits, this creates a hypervolume in d dimensions. Finally, if all dimensions are now normalized, this hypervolume reduces to an d-dimensional hypercube. Thus, any traditional design space which has independent variables that range continuously from lower limits to upper limits can be fully described by a hypercube of dimension d, where d is the number of design variables. This conclusion is important because if any unique d-dimensional design space can be reduced to a hypercube of dimension d, then any technique which draws samples from the space bounded by a d-dimensional hypercube, when properly scaled, can be used to draw samples in a similar fashion from the design space of interest. In this way, a sampling technique need not be developed for a particular problem, only for a generalized space, the d-dimensional hypercube.

The distinction of a hypercubic design space is important as many numerical and statistical sampling techniques (which were discussed in detail in the Background)

have been developed on the basis of sampling from this fundamental structure. Monte Carlo sampling, Taguchi methods and traditional Design of Experiments techniques all construct their set of designs to be sampled utilizing the full volume of the d-dimensional hypercube to which the design space for a particular problem is generalized [98, 22, 6]. These methods work well and will produce feasible results if the feasible design space is indeed hypercubic. However, what if the design space or rather the feasible portion of the design space is non-hypercubic? If this is the case, then some of the designs generated by a hypercubic-based sampling method will likely be sampling design points which are infeasible, that is they do not satisfy some constraint, or for whatever reason do not produce a successful result (code crash, failed convergence, numerical error, garbage result, etc.). As these designs are infeasible and/or failures, the computational effort expended in evaluating them can be considered wasted due to the absence of successful output. Now if the feasible design space is only a small perturbation from a hypercubic design space, or the analysis is rather computationally inexpensive these hypercubic sampling methods can still be used and a certain percentage of failures must be tolerated. But what if the feasible design space is a significant departure from a hypercubic space and the analysis of that space is computationally expensive. In this case, it stands to reason that hypercubic sampling methods will waste large percentages of their computational effort evaluating infeasible or unsuccessful designs. This result suggests that perhaps a different sampling method should be used, one that can recognize the feasible space is non-hypercubic and thus sample it more intelligently.

The physics-based computational design and analysis of an advanced aircraft concept has been established to be a computationally expensive problem, but is it non-hypercubic, and if so, how could that be determined, i.e. what are the characteristics of a non-hypercubic feasible design space? Expressing these concern more formally produces a set of research questions:

Motivating Question 2.1 (MQ2.1):

What characteristics make a feasible design space non-hypercubic?

Motivating Question 2.2 (MQ2.2):

Is the feasible design space for the physics-based computational design and analysis of an advanced aircraft concept non-hypercubic?

As the bounds of a hypercubic design space are defined strictly by upper and lower variable limits the design space is subject only to 2d active linear inequality or side constraints. However, the side constraints imposed to limit the range of the design variables considered are often not the only constraints applicable to a particular problem. Other constraints exist in the form of performance constraints, physical or kinematic constraints and non-physical constraints which arise due to the limitations of computational models. These constraints serve to limit and restrict the feasible design space, and while certain constraints like those related to performance may artificially erect boundaries in the design space, others demarcate regions of the design space which should not, or simply cannot be explored. For example, consider the design of a wing spar for a common cantilever wing and let the height of this wing spar be a design variable which can be increased to increase the bending rigidity of the wing. If the airfoil sections of the wing are also treated as design variables then there arises a physical constraint that limits the height of the wing spar to no more than the thickness allowed by the airfoil sections chosen. Unlike performance constraints, this constraint is unable to be compromised or perturbed as a design in which the wing spar protrudes from the wing OML is not only highly undesirable, but also physically impossible. Such a constraint could be represented as an equation of the form:

$$h_{spar} \leq \left(\frac{t}{c}\right)_{airfoil} * c \tag{9}$$

Equation 9 is a linear inequality constraint which is not parallel to any side constraints defining the design space limits and thus the boundaries of the hypercube. Because of this it eliminates a portion of the design space from consideration and in doing so transforms the feasible design space from a shape that can be generalized as a 2-D hypercube to a space that is non-hypercubic. Figure 3 illustrates how the feasible design space is impacted due to this physical constraint.

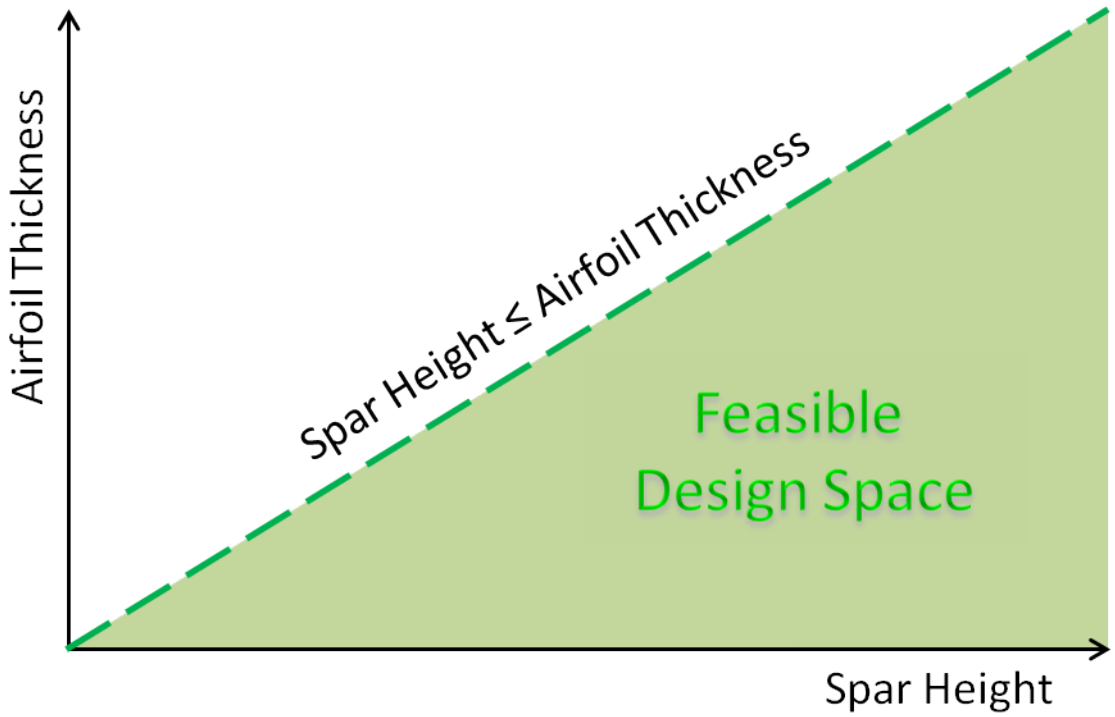

Figure 3: Non-Hypercubic Feasible Design Space Resulting from an Active Linear Constraint that is Non-Parallel to Design Space Side Constraints

Another example of the design space being transformed from a hypercube to a non-hypercubic feasible space through the application of a constraint can be seen when examining the limitations of a particular Vortex-Lattice analysis code called Athena Vortex Lattice AVL. While a very fast and flexible tool, like all other computational codes, it is not without limitations [19]. One particular limitation arises when considering flight conditions at high Mach numbers. Because AVL uses the

classical Prandtl-Glauert PG transformation to correct for compressibility effects, it is limited by the range of validity of this model. The PG transformation produces reasonable results up to a free stream Mach number of approximately 0.6, after which it breaks down as critical Mach numbers are approached in which the beginnings of shock formation can be observed on typical wing sections [5]. However, "for swept-wing configurations, the validity of the PG model is best judged using the wing-perpendicular Mach number" [19]. Where the wing-perpendicular mach number which the wing 'sees' can be calculated as:

$$M_{\perp} = M_{\infty} * cos(\Lambda) \tag{10}$$

Where M_{∞} is the free stream Mach number and Λ is the sweep angle of the wing. Since for a swept wing, the PG correction is limited by this wing-perpendicular mach number an interesting non-linear constraint which is a function of both free stream mach number and wing sweep arises to describe the region of validity of the computational model. Equation 11 provides an expression for this non-physical constraint (induced by computational model limitations) and when applied to the design space (considering variables Mach number and sweep angle) produces a non-hypercubic feasible space as seen in figure 4.

$$M_{\infty} \leq \frac{0.6}{cos(\Lambda)} \tag{11}$$

Based on these two examples an observation can be made about the nature of when a non-hypercubic design space can be expected to be present for a particular problem. As was seen with certain constraints which deny regions of the feasible space within the region defined by the variable limits, the feasible design space is altered from that of a regular hypercube. It is important to note that these constraints may be linear or non-linear in nature, but if linear must not be parallel to existing side constraints defined by variable limits. If parallel, the design variable ranges can

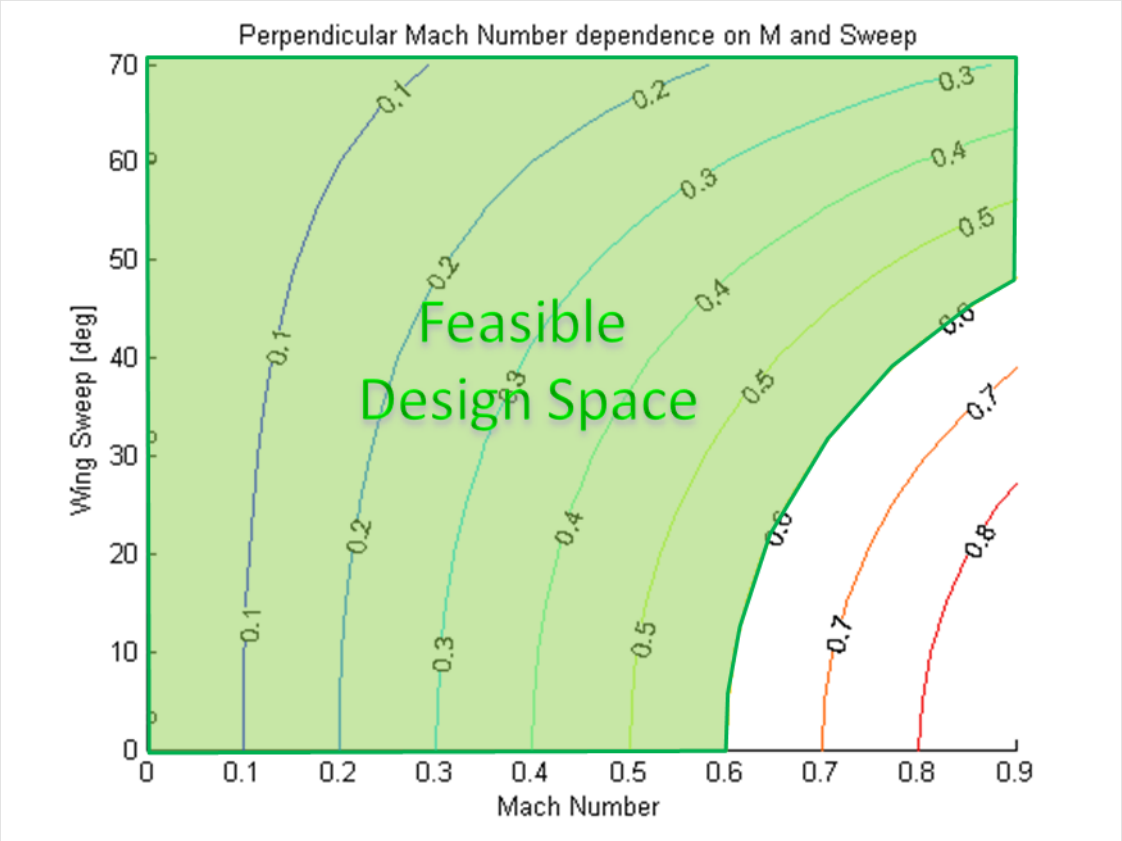

Figure 4: Non-Hypercubic Feasible Design Space Resulting from an Active Non-Linear Constraint

simply be redefined and the hypercubic shape preserved as a subset of the original design space. Furthermore, it is important that these constraints be active, i.e. exist within the bounds of the hypercubic design space, because if the variable ranges are restricted such that these constraints are no longer encountered, the space will likely remain hypercubic but sacrifice a significant amount of volume which was originally intended to be explored.

Observation 2.1.1: Regarding MQ2.1:

If one or more active non-linear constraints or active linear constraints non-parallel to existing design space boundaries are present in the design problem, then the resulting feasible design space will be non-hypercubic.

Caveat: Although performance related constraints contribute to the definition of the feasible design space, they may be somewhat arbitrary or variable in nature. When constructing regressions or surrogate models to represent a design space it may be beneficial to temporarily ignore these constraints, allowing the regressions constructed to remain valid and not subject to extrapolation should any performance constraint be perturbed.

Now, using the same logic which led to the previous observation, a similar result can emerge if the design space features significantly correlated design variables. In the traditional sense, design variables are intended to be defined such that they are entirely independent. If this assumption is true then neglecting the effects of all constraints and computational method limitations, an individual variable should be able to produce feasible designs when it assumes values throughout its range of consideration irrespective of the values of any of the other variables. However, if a variable at a particular value requires other variables to lie within a certain range (smaller than that given by the design space limits) in order to produce a feasible design, it can be said that these variables are correlated. If variables trend together then it follows that the cross-section of the design space over which they span will illustrate a similar trend. Finally, if an observable trend exists in the design space, then designs which do not obey this correlation yet exist within the bounds defined by the hypercube may be infeasible. If this is the case, then it can be hypothesized that sufficiently correlated design variables can cause a feasible design space to be non-hypercubic. As an example, consider two design variables which may, for computational simulation purposes, be desirable to be varied independently: wing taper and wing twist distribution. Wing taper is utilized to help shape the wing lift distribution and provide volumetric and structural advantages. It is desirable to have a low taper ratio to minimize wing weight and increase fuel volume for a given aspect ratio, however too much taper can result in the increased risk of wing tip stall [12, 78]. Wing twist is

typically used to achieve favorable stall characteristics such that during stall, flow remains attached over portions of the wing which contain control surfaces, ensuring control is maintained [12]. Changing the twist of a wing changes the local incidence and thus local effective angle of attack or a given wing section. Therefore in practice, to achieve similar stall characteristics, a wing with a lower taper ratio will require more washout (negative twist) at the wing tip than a wing with a less taper (higher taper ratio). Thus these variables although perhaps considered independent for computational simulation purposes are truly correlated and some regions of the design space will be denied, or made infeasible due to this correlation once these practical considerations are enforced. This chain of logic leads to the following observation:

Observation 2.1.2: Regarding MQ2.1:

If significant variable correlation exists between design variables over the ranges in which they are considered in the design problem, then the feasible design space is non-hypercubic.

Due to the nature of the specific problem considered, i.e. its high-dimensionality and coupled multidisciplinary nature, it is expected that this design problem will feature active linear and non-linear constraints as well as correlated design variables which all contribute to restriction of the feasible design space. However, it is not sufficient to simply assume a feasible design space will be non-hypercubic and thus potentially inefficiently explored by hypercubic based design space exploration techniques. Furthermore, knowledge about the characteristics or properties of the design space may not always be known a-priori (for example in the use of a 'black box' model or code). Thus it is highly desirable to have a means of hypercubic classification for a given design space, which, given only an initial design space sample output, could determine whether the feasible space was hypercubic or not. In regard to discovering a method suitable for hypercubic classification a formal question is posed to help guide experimentation:

Research Question 1 (RQ1):

How can the design space be classified as Hypercubic (HC) or Non-Hypercubic (NHC)?

2.1.3 Repeatedly Explored

Again revisiting the motivation and the overarching research question, the physics-based computational design and analysis of an advanced aircraft concept was also characterized by requiring repeated exploration. But why is this so, and what are the implications of this repeated exploration?

Motivating Question 3 (MQ3):

What makes the physics-based computational design and analysis of an advanced aircraft concept require repeated exploration and what consequences arise due to this requirement?

As posited by the previous section, active constraints which constrain the feasible design space can be expected for this and like problems. These constraints can form the boundaries of the feasible design space and make it non-hypercubic. However, some constraints, particularly those associated with performance, may not be static throughout the development of the technology and the associated aircraft system with which it is integrated. Complications arise due to the inherent variability of these constraints. This variability can be a product of factors such as: customer preferences, competitor actions, new regulations, volatility in fuel prices, cost overruns and schedule slippages. Combining these factors with the typical time-scale of large aircraft development programs often results in a final system that is noticeably different from what was originally envisioned [61]. This translates to a feasible design space in flux throughout the design process where the shape, characteristics and preferred regions of the design space may change drastically with time. Because of this, it is insufficient to explore and analyze such a design problem only once. As requirements

change, development progress is made and assumptions updated, so must the design space reflect this evolution.

Revisiting the design problem multiple times to address the evolution of assumptions and constraints comes at a cost. Specifically, analyses must be rerun to ensure changes are propagated. If the design space is revisited and re-explored in the same fashion for every update, it is very possible that regions of infeasibility will be continuously re-evaluated and the same waste of computational resources repeated again and again. Furthermore, the initial computational cost required to explore the design space is simply multiplied by the total number of times it must be explored. Form many problems, this cost may simply be tolerated, but for problems which are already computationally expensive, such a cost may not be acceptable. Furthermore, for problems which are non-hypercubic in nature, initial exploration will yield information about the characteristics of the feasible design space which can be used for future revisits (or to isolate and identify errors within the environment). Although some of the constraints and assumptions may change from iteration to iteration, it is likely that many will remain the same, thus in this situation it is beneficial to look to the past before going forward, to use the knowledge gained from previous explorations of the design space to guide new efforts. An analogy can be drawn by looking at the history of exploration of our planet. Explorers of old used cartography, the practice of map making, to record their discoveries for posterity and allow areas to be revisited with more knowledge each time. It is important to note that although these maps were by no means perfect, they served as a useful guide for future exploration in which they were revised and refined. Learning from this analogy, for a design problem which is both computationally expensive and non-hypercubic in nature, it is likely beneficial, i.e. will conserve computational effort, to generate an accurate (i.e. with enough resolution) map or bounding of the feasible design space to guide future exploration.

Observation 3: Regarding MQ3:

The repeated exploration of the design space for an advanced aircraft concept is required due to evolving constraints and updated assumptions associated with technology development. This repeated exploration results in an increase in necessary computational resource expenditure and the value of knowledge regarding the design space.

Given this observation regarding the characteristics of the problem and the assumption that some means of leveraging past knowledge about the design space will be helpful in its future exploration a formal research question regarding design space boundings is formulated:

Research Question 2: (RQ2):

Will an appropriate bounding constructed for a non-hypercubic design space generally provide an advantage for future exploration of the space?

2.2 Relevant Research Thrusts

As the overarching research objective of this work was to identify a methodology to enable general design space exploration for problems which feature the aforementioned challenges, three primary research areas were identified. These research areas are as follows:

- Design Space Exploration
- Design Space Classification
- Application: Conceptual Aircraft Design

For each of these three research areas, criteria were identified which were important to the construction of a decision support methodology for design space exploration. Additionally, specific alternatives identified through research were enumerated for these

criteria. This decomposition of research areas can be viewed in the morphological matrix presented in Table 2. This matrix served as a guide for the major thrusts of research within this work. In order to elicit the methodology sought by the ORO, alternatives were evaluated through literature review or experimentation for each of the necessary criteria.

2.2.1 Design Space Exploration

All design problems which involve computational analysis and do not have a direct or analytic solution must employ some form of numerical sampling that dictates which designs are evaluated and in what order. Design Space Exploration DSE is the systematic process of evaluating design alternatives spread throughout the design space [107]. There are many techniques that exist for sampling the design space and performing DSE. Selection of an appropriate technique is dependent on a number of factors such as [6, 68]:

- DSE objective:

  Compare the relative impact of design variables

  Screen to determine most important main effects

  Fit regressions or surrogate models

  Determine an optimum design

  Extract information for future DSE

- Design space characteristics (if known)

  Number of design variables/factors

  Continuous or discrete

  Constrained (besides variable limits) or unconstrained

  Correlation between design variables

Table 2: Morphological Matrix for Research Decomposition

Research Area	Criteria	Alternatives					
Design Space Exploration	Sampling Technique	Brute Force	Optimization	Pseudo-Random	Traditional DOE	Adaptive Sampling	
Design Space Classification	Hypercubic Classification	Visual Inspection	Pearson's r	Spearman's rho	Covariance	Mutual Information	
	Feasible/Infeasible Partitioning	Global Grouping	Set-Based Grouping				
	Feasible Space Bounding	Supervised Machine Learning Methods	Unsupervised Machine Learning Methods	Reinforcement Machine Learning Methods	Geometric Methods		
Conceptual Aircraft Design	Modeling and Simulation	Historical Data/Regressions	Analytical Models	Numerical Models			

Behavior at design space extremes

- Computational resource budget available
- Computational expense of design evaluation (function call)
- Amount of global exploration desired
- Batch execution of sample set desired

Certain DSE methods are more tailored to certain objectives and sampling design spaces with certain characteristics. By examining the existing techniques for DSE and keeping in mind the characteristics of the particular problem to be examined, the most suited technique for performing DSE for the physics-based computational design and analysis of an advanced aircraft concept can be identified.

Motivating Question 4 (MQ4):

For a problem which can be classified as computationally expensive, having a non-hypercubic design space and requiring repeated exploration, what are appropriate techniques for performing Design Space Exploration (DSE)?

2.2.1.1 Brute Force Sampling

Brute force or 'one variable at a time' sampling is conceptually the simplest technique for DSE. This method simply involves trying every combination of variables possible by changing the value of only one design variable at a time. This exhaustive form of DSE is: very simple to implement, applicable for all forms of design spaces whether continuous, discrete, constrained or unconstrained, excellent at global exploration and identification of optimum designs and will provide a great amount of data to which quality regressions can be fit. This applicability comes at a great cost however, as each design explicitly enumerated must be evaluated, the computational cost is

always proportional to the number of designs which can be considered. For practical problems this expense is most often unacceptable, and this technique is often only adopted when the experimenter is not limited by computational resources or the ease of implementation of this method outweighs the all other costs.

2.2.1.2 Optimization Based Sampling

Optimization based sampling is an umbrella given to all sampling techniques which rely on some objective function or design fitness to determine where the next samples in the space should occur. This set of DSE techniques are primarily concerned with finding a design (or set of designs) which is an optimum with regard to some predefined metrics or objective function. Optimization based sampling methods come in many types and can be either deterministic or stochastic in nature and often feature means of dealing with various types of constraints (however, these constraints must often be explicitly defined). Some examples of optimizer based sampling algorithms are: Grid Search, Steepest Descent, Powell's Method, Newton's Method, Genetic Algorithms, Simulated Annealing and Particle Swarm to name a few [101]. Since the objective of these methods is to efficiently find optimum designs, that is primarily where their strengths lie with regard to DSE. Depending on the method, they range from moderate to significant difficulty in implementation and while typically providing excellent exploration in neighborhoods around optimum designs (very good at exploitation) these methods rarely provide thorough global exploration. Even in cases when large portions of the design space are to be explored, the density of sampling is typically extremely non-uniform and concentrated near local optima. Because of this, global characteristics of the design space may be poorly understood as focus is placed near the local optima. Furthemore, using this method of sampling to then construct regressions results in models which are heavily biased/trained in regions of the design space near optima with poor representation of other areas more sparsely

sampled. Lastly, as optimizer based sampling is dependent on an objective function to explore the design space and discover optima, should this function change or constraints shift over the course of the design process, the best designs which were so laboriously sought and exploited may lose their optimality.

2.2.1.3 Pseudo-Random Sampling

Pseudo-Random Sampling or Pseudo Monte Carlo PMC is derived from traditional Monte Carlo Sampling and is essentially the numerical implementation of a pseudo-randomized design point [15]. "The prefix pseudo- refers to the use of a pseudo-random number generation algorithm that is intended to mimic a truly random natural process" [29]. In this method of DSE, each design is specified by design variable values which take on a pseudo-random value selected in between their lower and upper specified ranges. Sampling in this fashion is advantageous because given a random number generator, this method is very easy to implement. Furthermore, given enough samples, theoretically there is no biasing toward any region of the design space as every design point is just as likely (baring numerical errors) to be selected as any other. This trait makes this sampling method particularly well suited for cases in which little to nothing is known about the design space. However, in practice, if not enough samples are used to explore the design space, the distribution of design points may not be representative of true uniform distributions on each design variable. Because of the random nature of the selection of designs there is no formal structure or logic which governs how designs are placed in relation to another. For this reason, this sampling method makes it more difficult to extract sensitivities to any one variable as well as evaluate specific locations of interest in the design space, such as extremes.

2.2.1.4 Traditional Design of Experiments

"Design of Experiments was developed in the early 1920s by Sir Ronald Fisher at the Rothamsted Agricultural Field Research Station in London, England" [6]. Structured

statistical sampling techniques developed to optimize the information return from a limited expenditure of experimental resources, DOE have often been employed to efficiently extract information from large combinatorial spaces [22]. Many forms of DOE exist, ranging from the full and fractional factorial orthogonal designs to varied space filling designs [29, 15, 41, 82, 86, 91]. Some designs are intended to better capture behavior of the problem near the extremes of the design space, others closer to the interior. Some designs require a large or fixed number of samples while other specialize in utilizing relatively small or sample sets or arbitrary size [80, 102, 90]. Should some information about the problem be known, the various types of DOE allow the experimentalist many options in choosing the most appropriate DOE. Because of their structured nature, DOE are very capable in screening for design variables which contribute most the variability of responses. Furthermore, depending on their structure, DOE can allow statistical inferences to be made about design spaces and the variables they are composed of with relatively few samples. Additionally, DOE are often utilized to sample the design space in order to produce regressions or surrogate models for the response space. Certain DOE, such as Latin Hypercube Sampling LHS, have been shown to be very adept at providing a stratified and relatively unbiased sampling of the design space with a limited number of design cases enabling the production of regressions often superior to those obtained through PMC sampling [97]. DOE do have limitations however, and unlike PMC some designs can take significant computational expense to compute (determine where designs should be placed) [86]. Furthermore, traditional DOE are constructed under the assumption that the design space is a regular hypercube, should this assumption not be true, sample points may be placed in regions that are infeasible or will result in failure, and if so, certain desired properties like unbiased sampling or orthogonality can be lost.

Consequences of Violating Orthogonality Many traditional DOE utilize orthogonal structures in their distribution of designs throughout the design space. "An experimental design is orthogonal if the effects of any factor balance out (sum to zero) across the effects of the other factors" [69]. Orthogonality is a desirable property for experimental design as it allows for effects to be isolated and confounding between factors to be eliminated. With an orthogonal data set, techniques like the Generalized Method of Moments and Analysis of Variance $ANOVA$ can be used for parameter estimation and sensitivity analyses respectively. Additionally some regression techniques, such as Response Surface Methodology RSM, rely on orthogonal data sets to accurately estimate regression coefficients. Many DOE are not orthogonal however, either due to non-orthogonal construction (PMC, LHS) or failed data points which alter the structure of the DOE by denying any usable response values. The loss of orthogonality prevents proper the use of the aforementioned techniques and makes drawing statistical conclusion about the design space more difficult with a given set of data. However, depending on the ultimate goal of the design space exploration effort, the violation of orthogonality may not be all that consequential. "When setting up a computer experiment, it has become a standard practice to select the inputs spread out uniformly across the available space" [82]. Space filling designs are commonly considered most appropriate for computer experimentation and yet many are non-orthogonal. Some of the reasons for this are that space filling designs are able to better distribute cases throughout the space and are often not restricted to the specific case requirements orthogonal designs need to remain balanced. Furthermore, advanced regression techniques such as artificial neural networks ANN can achieve high accuracy for complex responses without requiring an orthogonal input space. For these reasons and the inherent fragility of orthogonal designs when subjected to a non-hypercubic design space, no orthogonal DOE will be utilized to examine design spaces within this work.

2.2.1.5 Adaptive Sampling

Adaptive sampling is an iterative form of sampling the design space. Unlike traditional DOE, or PMC, instead of seeding the design space with all its computational resources (available cases) at once, adaptive sampling uses information from previously evaluated design points to inform where the next set of sample points are to be placed [90]. In this way, adaptive sampling is somewhat similar to optimization based sampling, yet it differs in that it is not necessarily concerned with locating optimum values, but rather allocating samples to regions of interest (whatever that interest may be). Through iterative sampling of the design space, a feedback loop is created which allows for the more intelligent placement of samples. This feedback loop is especially useful if certain regions of the design space are infeasible as these regions can be avoided in future sampling iterations.

As the global feasible region for a general design space may not be hypercubic in nature it is prudent to perform adaptive sampling for DSE if possible. In this way, computational resources can be more efficiently allocated through multiple iterations to feasible regions as the boundaries of the design space become understood. Additionally, resources can be utilized to either explore or exploit knowledge gained through previous iterations of DSE allowing for the refinement of boundaries within the space as well as an increase in sample density in regions of interest. For these reasons, adaptive sampling was identified as the DSE technique most appropriate for the decision support methodology sought within this work.

2.2.2 Design Space Classification and Bounding

As suggested in the overarching research objective, an accurate bounding/representation for the design space is desired to enable the efficient analysis of problems like the physics-based computational design of an advanced aircraft concept. Based on

the aforementioned characteristics of this class of problems, a bounding of the design space, based on the classification of sample design cases evaluated throughout the space for their feasibility with regard to constraints and computational method limitations will likely prove helpful in efficiently addressing such problems. If this assumption is accepted, the questions still remain of how should designs be classified and this bounding be created? Posed more formally:

Motivating Question (MQ5):

For a problem which can be classified as computationally expensive, having a non-hypercubic design space and requiring repeated exploration, what are appropriate techniques for classifying and bounding the feasible design space?

Motivating Question (MQ5.1):

What is a suitable method for performing hypercubic classification for a general design space?

Motivating Question (MQ5.2):

How should individual designs be classified to best enable the boundings of the feasible design space?

Motivating Question (MQ5.3):

How should the boundings of the feasible design space be constructed?

,

2.2.2.1 Hypercubic Classification: Mutual Information

Motivated by RQ1 and MQ5.1 and the desire to find a robust method to provide hypercubic classification for arbitrary design spaces, multiple means of detecting features within a general design space were investigated. The simplest form of hypercubic classification can be performed through visual inspection of the design space.

Through examination of 2-D and 3-D cross-sections of a given design space, NHC regions can potentially be identified where there is separation of feasible and infeasible cases. However this means of classification is limited to dimensions less than or equal to three and thus cannot reliably identify all features which may yield a NHC feasible design space.

Other candidates for hypercubic classification involve examining the dependence or correlation between design variables when only feasible cases are considered. The design variables can be thought of as random variables whose discrete distributions are specified by the collection of unique values the variables take for the design cases specified in a given sample. For example, if the design space was sampled with PMC, the distributions for each of the design variables should be roughly uniform. Any features which would produce a NHC feasible design space would then manifest as some form of dependence or correlation between the design variable distributions for the feasible cases. With this realization, multiple correlation/dependence measures such as the linear correlation coefficient (Pearson's r), the rank correlation coefficient (Spearman's rho) and variable covariance were examined for their ability to detect the signs of a NHC feasible design space. Unfortunately these methods although relatively easy to implement, cannot detect all forms of correlation/dependence (particularly struggling with non-linear relationships) and thus the concept of Mutual Information MI from Information Theory was investigated.

Arising from information entropy estimation, MI is a measure of dependency between random variables. As Kinney explains: "Mutual information rigorously quantifies, in units known as 'bits,' how much information the value of one variable reveals about the value of another" [45]. MI is defined mathematically as follows [50]:

Let X and Y be two continuous random variables such that the marginal densities of

X and Y are $\mu_x(x) = \int dy\mu(x,y)$ and $\mu_y(y) = \int dx\mu(x,y)$, then MI is defined as:

$$MI(X,Y) = \iint dxdy\mu(x,y)log\frac{\mu(x,y)}{\mu_x(x)\mu_y(y)} \tag{12}$$

This expression for MI can be adapted for X and Y of discrete size (as would occur in design space exploration) by binning X and Y and approximating Eqn. 12 with the following finite sum [50]:

$$MI(X,Y) \approx MI_{binned}(X,Y) = \sum_{ij} p(i,j)log\frac{p(i,j)}{p_x(i)p_y(i)} \tag{13}$$

where $p_x(i) = \int_i dx\mu_x(x)$, $p_y(j) = \int_j dy\mu_y(y)$, and $p(i,j) = \int_i \int_j dxdy\mu(x,y)$ and $\int_i$ means the integral over bin i.

Eqn. Another useful quality of MI is displayed is elaborated by Kraskov:

> In contrast to the linear correlation coefficient [Pearson's r], it is sensitive also to dependencies which do not manifest themselves in the covariance. Indeed, MI is zero if and only if the two random variables are strictly independent [50].

MI is of special interest in regard to this problem because non-hypercubic features that may potentially exist in the feasible design space would affect the distributions with which the design variables are sampled. Not only should MI be able to detect correlation between design variables, but should a constraint deny a region of the design space in such a way to make the feasible space non-hypercubic, the absence of the infeasible designs will manifest as a change in the shape of the distributions in multiple design variables. Furthermore, in the case of a reduced hypercubic space existing within the original design space, the distributions on the design variables may not cover the full range, but their shape (if uniform) will remain unchanged from fthe original sample distributions. If these assertions hold true, then MI will not

only be able to detect non-hypercubic spaces in d-dimensions, but also distinguish them from reduced spaces which do not occupy the entire original hypervolume but still remain hypercubic. Because of these properties and advantages over existing correlation estimation methods, MI was investigated through experimentation for its potential to perform hypercubic classification.

2.2.2.2 Feasible/Infeasible Partitioning: Set-Based Design

Set-Based Design *SBD* or Set-Based Concurrent Engineering was investigated in response to MQ5.2. SBD is a design methodology which presents an alternative to traditional "point based concurrent engineering" [57, 54, 36]. Instead of locking in a large number of design decisions early on in the design process to move forward with an 'optimal' point-design, the methodology seeks to simultaneously increase knowledge about the design space and relevant requirements while maintaining as much design freedom as possible [63, 103]. Traditional SBD as depicted in Fig. 5, emphasizes bounding of the design space through the determination of feasible regions (sets) which arise from preferences in design variable values and analyses performed independently and in parallel by different disciplines [92, 31]. Because of their isolated nature and discipline driven preferences, these independently developed design spaces may be disjoint [96]. SBD then "integrates through intersection" to find a global feasible set with the objective to consider all constraints and "seek conceptual robustness" [92]. While this approach typically requires a much more resource intensive and lengthy conceptual design phase, it has been shown to mitigate costs and program delays associated with design problems encountered in detailed design or production. By carrying forward multiple design solutions further into the design process, SBD provides feasible alternatives when unforeseen problems render the nominal design infeasible or drastically degrade its performance [54].

As illustrated by the author in [48], using an adaptation of SBD, entire continuous

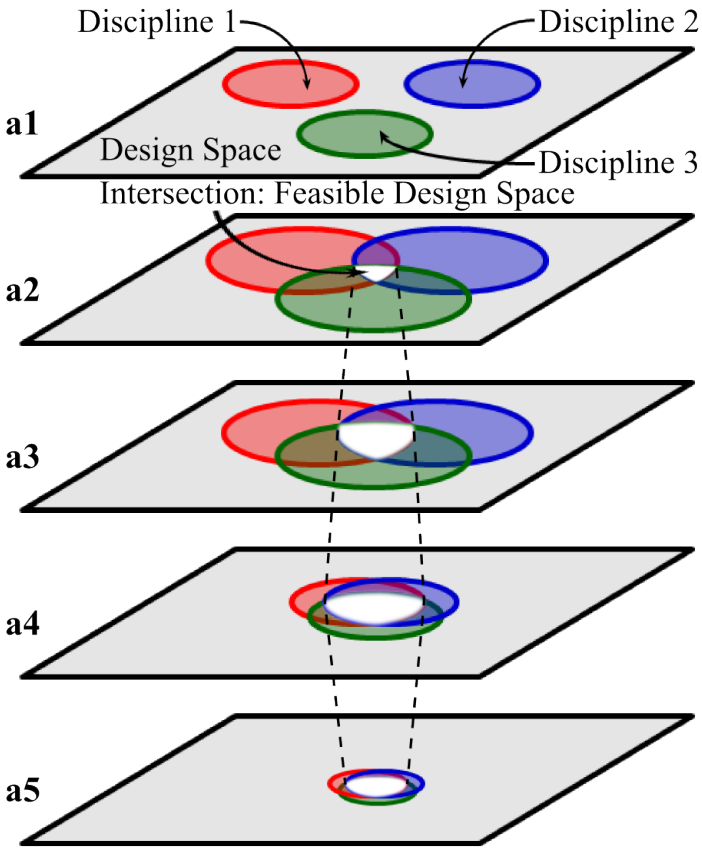

Figure 5: Graphical Depiction of Traditional Set-Based Design Methodology

regions of the design space (not just a set of multiple feasible points) can be bound through the determination of Constraint Defined Feasible Sets $CDFS$. A single CDFS represents the bounded region of the design space that is feasible with respect to an individual constraint, and when all are 'integrated through intersection' they yield the global feasible design space. Such an approach for classifying the design space is also advantageous in that it allows for a much more robust representation of the global feasible design space. Should an individual constraint or specific module within an environment be updated, only those CDFS affected by such a change need be reformed to again yield the global feasible design space. For problems subject to repeated exploration it is hypothesized that a SBD approach is especially advantageous over

a global approach to ultimately reveal the global feasible design space. Based upon this a formal research question is formulated:

Research Question 3 (RQ3):

If a bounding for the global feasible design space is desired, is it more effective to simply construct a global bounding fit to all NHC variables or construct this bounding through the intersection of Constraint Defined Feasible Sets $CDFS$ found by individually bounding each separable constraint?

2.2.2.3 Feasible Space Bounding: Machine Learning Techniques

Driven by MQ5.3, Machine Learning techniques were found to be a set of approaches for both classifying samples within a design space and then attempting to identify separate regions within these spaces (effectively bounding them) [13, 9]. "Machine learning is the body of research related to automated large-scale data analysis" [9], often also synonymous with Pattern Recognition, it is concerned with the development and use of algorithms which can learn and draw observations from data. Machine Learning techniques can typically be binned into one of three major categories (referring to the mechanism through which they learn from data): Supervised Learning, Unsupervised Learning, and Reinforcement Learning. Supervised Learning methods utilize a training data set in which a truth model provides the correct output/classification for this data set (i.e. the training set can be trusted to be correct). Unsupervised Learning methods attempt to learn from data without this verified truth training data set while Reinforcement Learning methods "are concerned with with the problem of finding suitable actions to take in a given situation on order to maximize a reward" [13]. For the design problem to be considered and problems of its class, although the explicit form of all constraints and boundaries of model limitations may not be known a-priori, designs will be able to be classified by whether or not they

satisfy certain constraints and/or cause the computational method to fail. With this information available for each design evaluated, Supervised Learning methods are the most appropriate to provide competing methods for the classification and bounding of the design space.

While there are many potential algorithms, based on the characteristics of the design problem and their popularity, success and applicability to the problem considered the following two machine learning techniques were considered as potential methods for generating an accurate and efficient design space representation [13, 9, 21]:

- Random Forests RF
- Kernel-Based Support Vector Machines SVM

It is important to note that although these two particular techniques were chosen for further investigation and use within experimentation, they are by no means the only types of supervised machine learning techniques applicable within the general methodology sought. It has been illustrated that the optimal machine learning technique varies with problem characteristics and no one technique is globally optimal [49, 21]. However, the two techniques chosen have also been shown to produce fairly robust results for a variety of data sets and were thus deemed sufficient candidates to examine the efficacy of utilizing supervised machine learning within the methodology [21].

Random Forests Random forests RF or random decision forests encompass a supervised machine learning technique which shows promise for the classification of design spaces likely to be encountered within this scope of work. This technique has been described in literature as being a good choice of classifier algorithm for generic problems where little is known a-priori [21]. Random forests utilize an ensemble method to average classification results produced by many (a forest) of classical decision trees trained via 'bagging'. 'Bagging' or bootstrap aggregating is a method

of selecting multiple training subsets from a set of data by sampling the original set uniformly and with replacement [14]. By constructing training sets in this way for each of the individual trees which compose the forest, the overall classifier is much less sensitive to an individual tree which may biased by an outlier within the data set, thus bagging is said to produce more accurate classifiers while avoiding overfitting [35].

An example of two separate decision trees trained through bagging is shown in fig. 6. From the original data set (featured in the center) two training samples are assembled through bagging and used to grow two distinct decision trees. Each node represents a decision made on a design variable value at which the decision tree has subdivided the space. Branches are then grown into each of those regions and the subdivision process continues until a branch has isolated a single class. Ultimately the space will be segmented into a series of reduced hyper-rectangles each approximately containing a unique class. These trained decision trees can now be individually queried with new points to determine with which class these suggested points likely belong. In a random forest, the predictions of all the decision trees are then averaged to determine what the forest's prediction should be for the new sample.

Random forests in addition to being relatively easy to understand and interpret, also have some useful properties when examined for their ability to construct classifier based boundings for design spaces. Firstly, a decision tree algorithm (and random forest by extension) isolates unique data classes by constructing hyper-rectangular cells within the space. Because of this inherent geometry, should the feasible design space be a reduced hypercube or a series of reduced hypercubes with respect to the original hypercubic volume, then a random forest classifier should be able to fit it very accurately. Perhaps more important is the ability of random forest classifiers to determine the relative importance of the different variables within the space. Because split nodes in the decision trees are defined by a particular variable and value, the

influence a given variable has over the forest predicting the correct classification can be observed through the comparison of results obtained with the inclusion and exclusion of the variable. If the predictive capacity significantly improves with the consideration of a certain variable then it is deemed important while if its inclusion in the model does little to change accuracy, it is likely unimportant (or its effect is confounded with another variable(s)). Variable importance can also be captured by examining the Gini impurity associated with the model which essentially measures the likelihood of misclassification due to nodes containing more than one class. If the inclusion of a certain variable allows for a significant decrease in Gini impurity then this means the random forest is able to subdivide the space into sets which are more homogeneous and thus provide a more accurate division of the classes. With $p(i|j)$ representing the proportion of the samples that are members of class c for a particular node j, then the Gini Impurity can be expressed by Eqn. 14. These capability could potentially prove very useful in not only reducing model complexity through ranking of important variables, but allowing for the identification of NHC design variables.

$$I_G(j) = \sum_{i=1}^{c} p(i|j)(1 - p(i|j)) \tag{14}$$

Random forest classifiers were also chosen for use within this work due to the existence of well developed tools and support for their implementation. The 'randomForest' R package with all of its built in functionality was an enabler for the rapid construction, analysis, and use of random forest classifiers utilized within this work [83, 23].

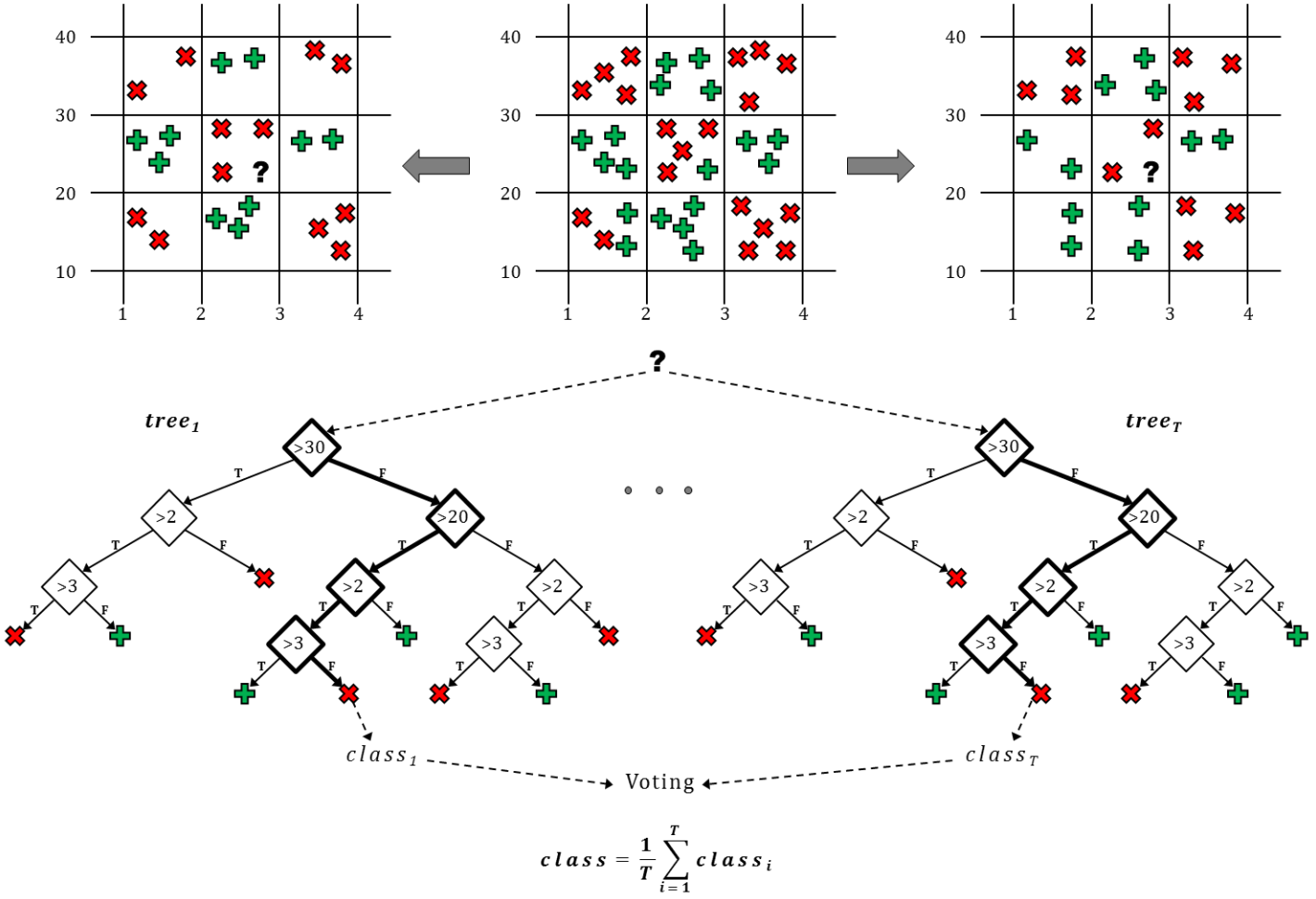

Figure 6: Random Forest Classifier Composed of Multiple Individual Decision Trees

Kernel-Based Support Vector Machines Kernel-Based Support Vector Machines SVM are another supervised machine learning technique which were considered apt for the classification of design spaces likely to be encountered within this scope of work. Kernel-based SVM construct classification boundaries by mapping the training data in the input space to a higher-dimensional feature space through the use of kernel functions and associated hyper-parameters. Using a kernel function (applying the 'kernel trick') allows the inner products between the images of all pairs of data to be calculated directly in the feature space, without requiring the explicit mapping to the higher-dimensional feature space [49]. There are a few different kernels typically associated with SVM but perhaps one of the most popular is the gaussian radial basis function RBF kernel function:

$$K(x, x') = exp\left(- \frac{\|x - x'\|^2}{2\sigma^2} \right) \tag{15}$$

SVM classifiers attempt to separate the data of different classes by as much distance/margin as possible in the feature space. The support vectors are the boundary points in the higher-dimensional feature space which are closest to those of the other class and thus determine the position and orientation of the separating maximum-margin hyperplane. By using kernel functions to map the original space to one of higher dimension, the potentially non-linear boundary in the input space can be coerced to linearly separable sets in the feature space. Expressed generally in eqn. 16, a kernel-based SVM is trying to maximize the following Lagrangian (L_P) with respect to $\overrightarrow{\mathbf{w}}$ and b (which together define the maximum-margin hyperplane).

$$L_P \equiv \frac{1}{2}\|\overrightarrow{\mathbf{w}}\| - \sum_{i=1}^{N} \alpha_i y_i (\overrightarrow{\mathbf{x_i}} \cdot \overrightarrow{\mathbf{w}} - b) + \sum_{i=1}^{N} \alpha_i \tag{16}$$

Where N is the total number of training samples, the $\alpha_i > 0$ are the 'support vectors', $\overrightarrow{\mathbf{x_i}}$ the individual samples and $y_i = 1$ or -1 designating the class to which the ith sample belongs [49, 106]. Figure 7 depicts the process of applying a kernel function to map input data to the feature space and then determine the maximum-margin hyperplane. The support vectors can be identified as the points in the feature space which anchor/define the planes offset from the maximum-margin hyperplane. Through the 'kernel trick' and utilizing the support vectors and margin maximization to construct the dividing hyperplane and thus determine the classification boundary, SVM methods do not suffer from the curse of dimensionality. Thus these methods can be applied to classification problems of very high dimension without becoming computationally infeasible.

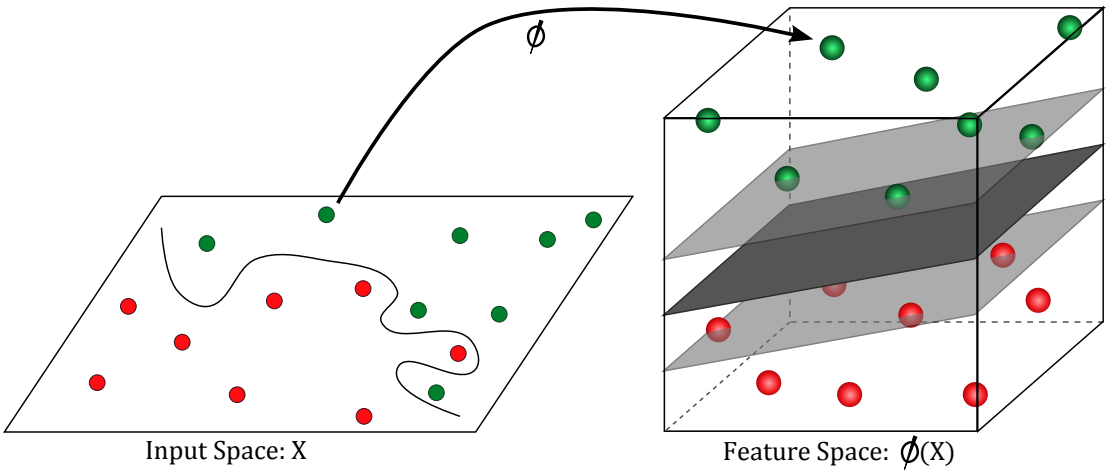

Figure 7: Depiction of Kernel-Based Support Vector Machine Mapping from the Input Space to the Feature Space and Constructing Separating Hyperplane

Kernel-based SVM classifiers were also chosen for use within this work due to the existence of well developed tools and support for their implementation. The kernlab() R package with all of its built in functionality was an enabler for the rapid construction, analysis, and use of SVM classifiers utilized within this work [83, 3].

Geometric Bounding Techniques: Delaunay Triangulation Again spurred by MQ5.3, to provide another option to bound the sets formed by a SBD approach which would be "integrated through intersection" to produce a bounding of the feasible design space, different geometric volume bounding methods were investigated. The field of computational geometry can be investigated to yield candidate methods and upon inspection, a few relatively simple techniques for bounding volumes can be seen to emerge [79, 16, 64]. These techniques vary in both complexity and accuracy for the representation of the hypervolume which they attempt to enclose, with the general trend that the more complex/faceted and specifically oriented the bounding volume, the more expensive it is to compute. A Delaunay Triangulation *DT* based method developed by the author was investigated but ultimately found largely infeasible for classes of problems in moderate to high dimension.

A Delaunay Triangulation *DT* in d-dimensions for a set of points P is a triangulation in which no point within the set exists inside the circumhypersphere defined by the d+1 points constructing a given simplex within the triangulation [79]. The Delaunay Triangulation is distinct from other traingulation methods in that it seeks to minimize the aspect ratios of its simplices by maximizing the minimum angle. Often used to build meshes in finite element methods, DTs provide a geometric network between all points within a given sample. Furthermore because their construction is geometric in nature, the DT is formed in the same space in which the points are specified (unlike with kernel-based SVM) thus preserving the physical meaning of design variables. Additionally due to the structure provided with DTs, adaptive sampling can leveraging the simplicies, and the relative size of simplices can be used to target regions of reduced density.

A DT based set bounding method which performed adaptive sampling through partial barycentric subdivision was devised and investigated by the author within this

work. It was found to provide adequate results for bounding feasible sets in low dimension, but when scaled to moderate to high dimensions was found computationally infeasible with resources available. While excellent for visualizations in low dimensions and constructing actual boundaries for the feasible design space, this method was ultimately abandoned in favor of more robust approaches. Figure 8 illustrates the DT method created by the author being used to bound and adaptively sample a NHC design space in 2 dimensions.

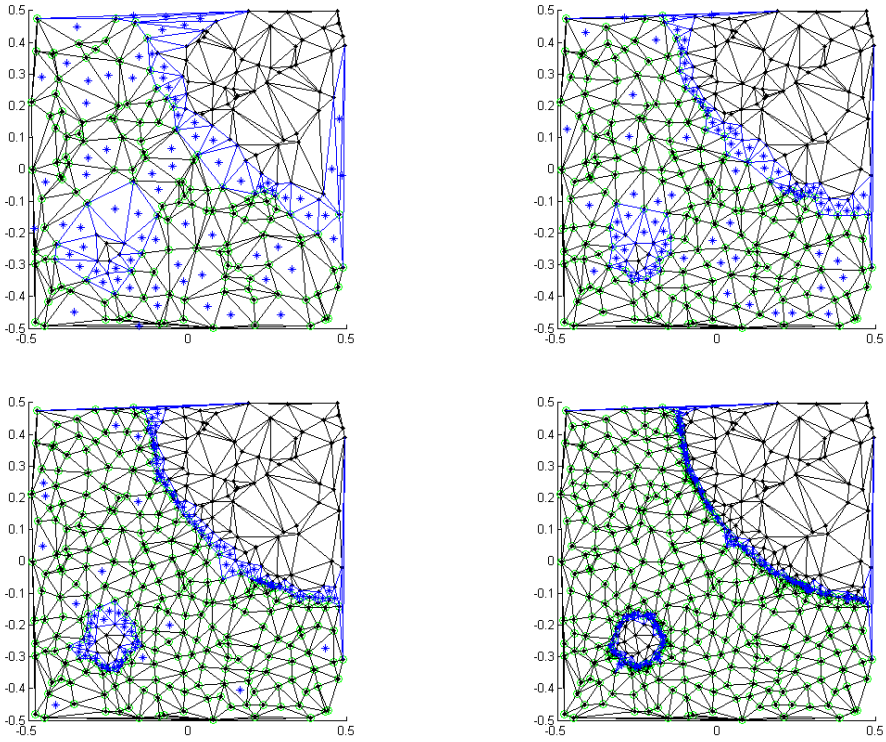

Figure 8: Adaptive Delaunay Triangulation Based Bounding

2.2.3 The Application: Multidisciplinary Conceptual Aircraft Design

The methodology sought in this thesis is motivated by a desire to more efficiently enable multidisciplinary conceptual aircraft design. In order to capture all the relevant physics as well as supply the designer with sufficient degrees of freedom, a

very large design space consisting of many variables is often desired. While this allows many factors and details to be considered within the conceptual design process, if historical regressions must be abandoned in favor of physics-based computational models (needed to evaluate advanced designs with emerging technology), then such dimensionality can lead to high resource cost. Without compromising the fidelity of computational environments, this thesis seeks a methodology that will allow such a design space to be thoroughly explored without wasting resources, irrespective of what constraints, correlations or regions of infeasibility lie within. Formally stated as a research question:

Research Question 4 (RQ4):

Can use of the proposed methodology demonstrate an improvement in efficiency and knowledge gain with respect to state of the art practices in design space exploration techniques for a realistic aircraft conceptual design problem?

2.2.3.1 Challenges

Before the advent of modern computational methods, aircraft conceptual design was largely performed using regressions against historical data [84]. Certain parameters would be regressed against vehicle reference weights or areas and collected with other aircraft of a similar type. These methods proved very useful and accurate as long as materials and manufacturing techniques marched along at an evolutionary pace. But, with the infusion of emerging technologies and the consideration of advanced airframe concepts such as the HWB, there is no historical data to rely upon for performance analysis.

To enable the design and analysis of advanced concepts a new paradigm was forged which embraced physics-based computational methods as replacement for the historical regressions. These physics-based methods come in two forms: analytical

models (which can be solved exactly) and numerical models. Analytical models are attractive due to their exact solutions, however often lack sufficient fidelity to model all relevant physical phenomena present within practical engineering problems. Thus, numerical methods are frequently utilized to address this gap, yet these methods came with their own limitations including computational expense, convergence issues and model error. Additionally the linking of different stand-alone codes and creation of so-called 'black box' tools where the user cannot access the source code can lead to many computational and user errors which produce failures. These errors can range from such things as embedded solvers that become ill conditioned and cannot converge to codes that are extended beyond the realm of their original application and thus produce unvalidated/garbage results if any at all. Therefore care must be taken when exploring computational design spaces, for while enabling large conceptual explorations, they are still bound by the laws of physics and the assumptions used in their creation.

2.2.3.2 Numerical Modeling: The Environmental Design Space

As moderate to high fidelity analysis was deemed necessary for the physics-based conceptual design of an advanced aircraft such as the HWB and therefore, a numerical method appeared most appropriate to accurately capture the physics present for such an application. The Environmental Design Space *EDS* is a numerical modeling and simulation environment composed of an integrated set of NASA developed computational models [46]. EDS was developed for the FAA to ultimately help predict and project emissions estimates for current and future civil transport aircraft [47]. EDS takes an approach similar to those followed in [24, 25, 53, 67, 18, 34, 85, 10] with the integration of various disciplinary analyses into a singular modular computer program. EDS is composed of the following main computational models which capture relevant physics from the different disciplines:

- Numerical Propulsion System Simulation $NPSS$
- Compressor Map Generator $CMPGEN$
- Weight Analysis of Turbine Engines $WATE$
- Flight Optimization System $FLOPS$
- Pressure and Temperature Correlations $P3T3$
- Aircraft Noise Prediction Program $ANOPP$

The modules composing EDS are integrated into a unified environment using the NPSS programming language to enable input/output I/O between modules. Figure 9 displays how data propagates through EDS.

Furthermore, as Gatian [26] states:

> EDS has been validated and calibrated using existing vehicle data for a wide variety of aircraft architectures and seat classes. Its capabilities have been proven through its application to various assessments for NASA, the FAA, and academia. The results of many of these studies have been presented at various conferences and published in leading aerospace journals. [17, 30, 32, 38, 40, 39, 43, 44, 65, 73, 74, 75, 77, 76, 87, 88, 89]

Based upon the physics-based capabilities of EDS for use in aircraft conceptual design, its established pedigree, and ability to be parallelized to reduce user execution time, it was selected as the modeling and simulation environment with which to benchmark the proposed methodology through experimentation.

2.2.4 Refined Scope of Research

Through literature review and preliminary testing of competing alternatives, the most promising techniques were identified for the proposed methodology. Some of these

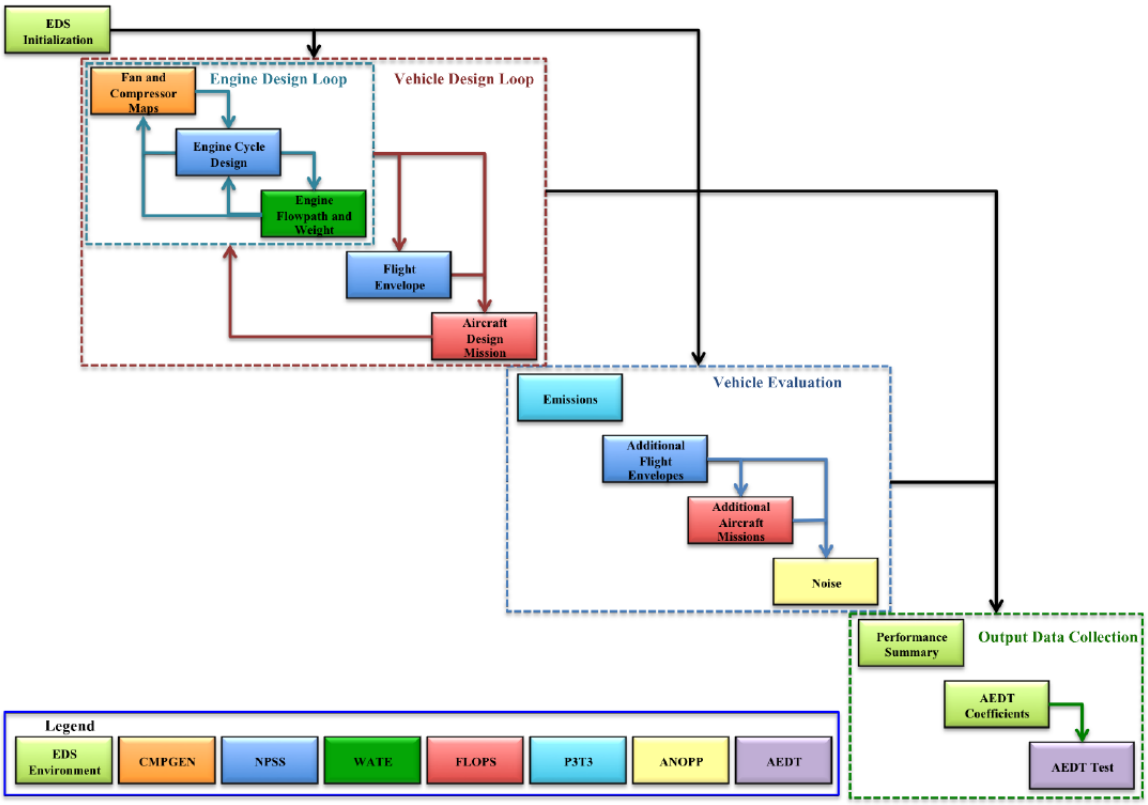

Figure 9: Environmental Design Space Module Layout [26]

techniques were able to be down-selected purely from conclusions drawn from synthesis of literature, while others require further experimentation to demonstrate their applicability or utility within the methodology. Table 3 illustrates the alternatives which were down-selected for use and experimentation within the proposed methodology.

Table 3: Down-Selected Morphological Matrix for Research Decomposition

Research Area	**Criteria**	**Alternatives**				
Design Space Exploration	Sampling Technique **(Lit. Synthesis)**	Brute Force	Optimization	Pseudo-Random	Traditional DOE	**Adaptive Sampling**
Design Space Classification	Hypercubic Classification **(EXP:1)**	Visual Inspection	Pearson's r	Spearman's rho	Covariance	**Mutual Information**
	Feasible/Infeasible Partitioning **(EXP:3)**	Global Grouping	**Set-Based Grouping**			
	Feasible Space Bounding **(EXP:2-3)**	**Supervised Machine Learning Methods**	Unsupervised Machine Learning Methods	Reinforcement Machine Learning Methods	Geometric Methods	
Conceptual Aircraft Design	Modeling and Simulation **(EXP:4)**	Historical Data/Regressions	Analytical Models	**Numerical Models**		

CHAPTER III

PROBLEM CHARACTERIZATION

3.1 Canonical Example Problem Description

The canonical example problem was meant to illustrate the presence of non-hypercubic design spaces in relevant aircraft design problems. A multidiciplinary conceptual wing design problem was selected for investigation in this example due to the inherent presence of requirements and constraints, from multiple physical diciplines, which when simultaneously considered would likely result in an NHC feasible design space. For this example, A simple physics-based environment was constructed for the design of a clean sheet wing with an integrated variable camber trailing edge $VCTE$ flap. Using Euler Beam Theory as a structural model and Prandtl Lifting Line Theory for an aerodynamics model [28, 12], a simplified wing can be designed and the simulated application of a variable camber trailing edge device evaluated. This choice of models is particularly advantageous because even though many assumptions need to be made and many finer elements of the wing structural and aerodynamic design omitted (such as complex wing box construction, sweep, compressible flow regimes), these models are sufficient to demonstrate some of the major physical interactions at work in a multidisciplinary aircraft design problem and highlight some of the challenges which make the exploration of such a design space difficult. Furthermore, they can provide an analytic solution for the multidisciplinary problem thus allowing for relative ease in querying and extracting characteristics of the design space. For simplicity and ease of design space visualization, the wing designed with this canonical example, as seen in Fig. 10 will feature only five design variables and will be restricted to a straight un-tapered wing with no twist or dihedral.

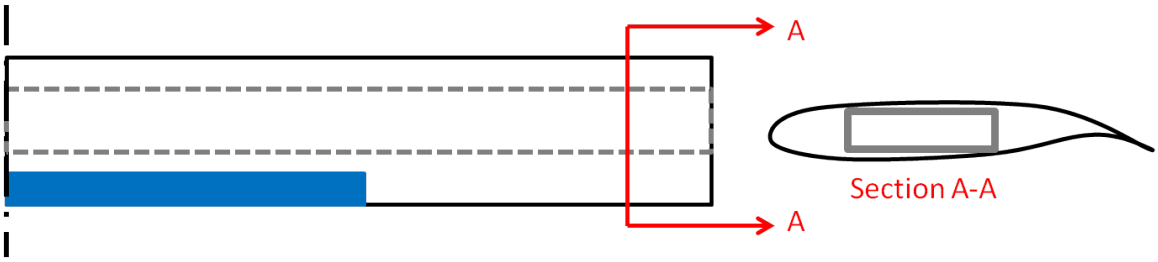

Figure 10: Planform and Cross-Sectional Views of the Wing Utilized for the Canonical Example Problem

3.1.1 Model Construction

The computational model constructed for the canonical problem consists of a coupled Prandtl Lifting Line Theory function and a set of expressions defined by Euler-Bernoulli Beam Theory. The Lifting Line function first discretizes the wing as a series of discrete horseshoe vortices emanating from the wing quarter-chord and distributed spanwise with a cosine spacing. These vortices produce a circulation given by equation 17. The Biot-Savart law is then used with the circulation distribution to determine the downwash induced in the wake of the wing (Eqn. 18). This result can then be combined with a definition of the local circulation as a function of the airfoil 2D lift-curve slope and resultant angle of attack (Eqn. 19) to produce the the monoplane equation (Eqn. 20) which provides a means to solve for the Fourier coefficients (A_n) for each of the horseshoe vortices. Finally the lift distribution can be obtained by the integration of the circulation distribution over the wingspan (Eqn. 21). This lift distribution is then used to provide the aerodynamic load used to size the required wing structure. To simulate the effect of a deflected VCTE, a flap section is specified over a portion of the wing span by adjusting the local zero lift angle of attack for that particular section. Figure 11 illustrates example lift distribution results with the left image depicting the lift distribution resulting from a nominal wing while the right shows the lift distribution which results with the addition of a trailing edge flap which spans from the wing root to approximately 60 percent of the total semi-span.

$$\Gamma(y) = \Gamma(\theta) = 4sV_\infty \sum_{n=1}^{\infty} A_n sin(n\theta) \tag{17}$$

$$w_i = V_\infty \sum_{n=1}^{\infty} \frac{nA_n sin(n\theta)}{sin(\theta)} \tag{18}$$

$$\Gamma = \frac{1}{2} a_0 V_\infty c(\alpha - \alpha_i + \theta_t - \alpha_0) \tag{19}$$

$$\sum_{n=1}^{\infty} A_n sin(n\theta) \left(sin(\theta) + \frac{na_0 c}{8s} \right) = \frac{a_0 c}{8s} sin(\theta)(\alpha + \theta_t - \alpha_0) \tag{20}$$

$$Ł = \rho V_\infty \int_{-s}^{s} \Gamma * dy \tag{21}$$

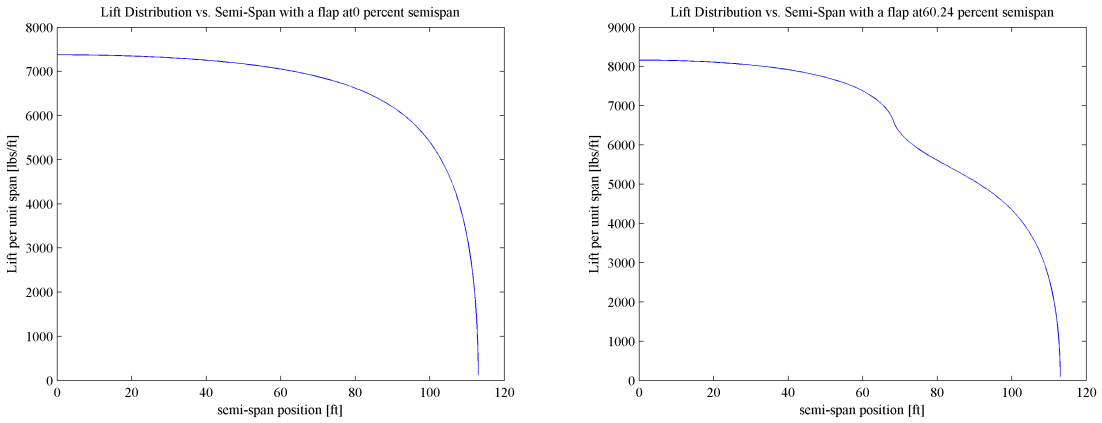

Figure 11: Example Lift Distributions Plotted against the Wing Semi-span

Once the aerodynamic load for a specified flight condition is calculated using the Lifting Line function, this load distribution can be combined with others such as the weight of the fuel stored inside the wing and the weight of the wing itself. This wing load profile is then completed with the addition of a point load representing the weight of an engine attached to the wing. Now with all the loads known, Euler-Bernoulli Beam Theory (Eqn. 22) can be applied and simplified for a beam of constant stiffness

to determine the distribution of shear forces (Eqn. 23), bending moment(Eqn. 24), beam slope (Eqn. 25) and beam deflection (Eqn. 26) in the bending plane.

$$\frac{d^2}{dx^2}\left(EI\frac{d^2w}{dx^2}\right) = -q(x) \tag{22}$$

$$V(x) = \int -q(x)dx = EI\frac{d^3w}{dx^3} \tag{23}$$

$$M(x) = \int V(x)dx = EI\frac{d^2w}{dx^2} \tag{24}$$

$$S(x) = \int \frac{M(x)}{EI}dx = \frac{dw}{dx} \tag{25}$$

$$w(x) = \int S(x) \tag{26}$$

Using this model and a critical flight condition or maneuver, a simplified wing can be designed with a set of design variables which provide details about wing aerodynamic and structural geometry. The performance of this particular wing can also be tracked through metrics of interest such as the lift-to-drag ratio, and the feasibility of the design assessed through the application of relevant constraints. Ultimately, after the application of constraints, feasible sets can be constructed and through plotting these sets of designs the shape and characteristics of the design space visualized.

3.1.2 Assumptions

In order to utilize such simplistic models for the complex coupled aero-structural wing design problem and still produce credible results and trends, many assumptions must be made. Some assumptions arise from the physical limitations inherent in the models themselves, while others are made for convenience and simplification of the analysis. Assumptions made for this canonical problem are as follows:

- General Assumptions

    1. Aircraft characteristics are representative of a 300 passenger civil transport with a total weight of 600,000 lbs and a fuel weight of 300,000 lbs
    2. Wing sizing maneuver is a +2.5g pull-up performed at an altitude of 39,000 ft and a Mach number of 0.84

- Aerodynamics Assumptions

    1. Limitations of Lifting Line Theory apply
        - Steady flow
        - Inviscid flow
        - Incompressible flow
        - High Aspect Ratio, unswept wings
        - Linear lift curve slope (a_0)
    2. Wing is modeled as the only lifting body

- Structural Assumptions

    1. Wing is idealized as a box beam of constant cross-section (constant moment of inertia along the span) and uniform wall thickness
    2. Wing is in pure bending and remains in the elastic region
    3. Wing material has the properties of solid Aluminum 7075-T6
    4. Wing weight is treated as a uniform distributed load across the span and assumed to be 1.8 times the weight of the box beam which comprises the primary structure
    5. Wing fuel weight is modeled as a uniform distributed load across the span and 60 percent of the total wing fuel is assumed to be stored within the cavity in the wing box beam

6. Engine weight is modeled as a point load on each wing located two fuselage diameters (approximately 40 ft) from the wing centerline

- Geometric Assumptions

1. Straight, planar, un-tapered, un-twisted, cantilever wing with constant airfoil section (except for VCTE segment)
2. Airfoil characteristics based on BAC 1 supercritical airfoil
3. VCTE flap span begins at the wing root
4. VCTE deflection is modeled as a -6 degree change in α_0
5. Box beam height is 80 percent of maximum wing airfoil thickness and width is 40 percent of wing chord
6. Useable fuel volume within the wing box is 90 percent of the cavity volume

Furthermore, the extent of the design space can be artificially reduced for visualization and analysis simplification purposes. For this canonical problem five wing design parameters deemed particularly relevant and familiar were and appropriate ranges for consideration were chosen, they are:

- **b**: wing span, Range: 160 - 260 ft
- **AR**: Aspect Ratio, Range: 8 - 12
- **t/c**: thickness to chord ratio, Range: 0.08 - 0.18
- **d**: beam wall thickness to box beam height ratio, Range: 0.005 - 0.200
- **sVCTE**: span of the VCTE flap section to semi-span ratio, *Range 0.1 - 0.8

*Note: for each design, a corresponding wing without an VCTE flap section was constructed for comparison

With these five design variables and the assumptions listed above, the wing can be fully defined and analyzed to determine how a particular design performs and what constraints it satisfies. By utilizing a DOE, combinations of design variable values can be specified for unique designs spread throughout the design space. The performance for each of these designs can be recorded and they can be classified as feasible or infeasible. With many constraints to satisfy, a wing design will only be considered globally feasible if it simultaneously satisfies all constraints.

3.1.3 Constraints

For any wing design to be considered in earnest and designed to a specific set of requirements, it must be subject to a set of relevant constraints. In this canonical problem, seven constraints are selected to represent some of the challenges encountered in wing design and hold the designs to a degree of realism. The constraints attempt to provide a small sample of similar aerodynamic, structural, aeroelastic, geometric and operational constraints that a wing designed using a more rigorous process would also feature. The constraints applied for this canonical problem are as follows:

1. **Wing Stall**: the wing angle of attack must be less than 16 degrees

2. **Maximum Moment**: the stress produced by the maximum bending moment encountered in the wing beam must not exceed the maximum allowable stress determined by the structure, material properties and a safety factor of 1.5

3. **Maximum Shear**: the stress produced by the maximum shear force encountered in the wing beam must not exceed the maximum allowable stress determined by the structure, material properties and a safety factor of 1.5

4. **Maximum Deflection**: the wing tip must not have a deflection magnitude

that exceeds 30 percent of the wing semi-span (to prevent the need for consideration of dynamic aeroelastic effects)

5. **Fuel Volume**: the internal wing box volume available for fuel storage must be at least equal to that required

6. **Wing Weight**: the wing weight must not exceed 60 percent of the combined empty and payload weight

7. **Minimum Gauge**: the wing box wall must have at a thickness of least 0.032 inches

By sampling the design space with a DOE and then applying the listed constraints for each design, sets of designs with common characteristics can be constructed. Some designs may not satisfy any of the constraints, these designs can be classified as **globally infeasible**. Many designs satisfy at least some of the constraints and thus for those particular constraints these designs are classified as **locally feasible** and can be grouped to form constraint defined feasible sets. The few designs which happen to satisfy all constraints are classified as **globally feasible** and the subset of the design space to which they belong defines the extent of the feasible design space.

3.2 Results

The canonical example problem, while a reduced and less realistic investigation compared to practical design problem, incorporated the effects of some of physical disciplines inherently present in aircraft conceptual design. As such, the results of this example can be envisioned as a subset of the potential phenomena present in a realistic design problem. The results for this example problem were largely analyzed through visual inspection, yet even viewing 2 and 3 dimensional projections of the design space provided significant evidence that the global feasible space exhibited non-hypercubic characteristics.

3.2.1 Design Space Visualization

To gain understanding and better extract defining characteristics of the problem, both the design space and the response space can be visualized. The response space provides a depiction of the performance of the designs considered and can ultimately be utilized to determine the optimal designs. Figure 12 illustrates the shear force and moment experienced by the wing. The asterisks represent the forces and moments from various load sources for a nominal wing, while the solid lines belong to a wing with a simulated VCTE flap, the black lines represent the superposition of all contributions. Figure 13 shows the wing deflection response. Figures 14 and 15 depict the 5-D design space in 2 and 3 dimensions showing designs which satisfy the fuel volume constraint in green and those satisfying the wing weight constraint in red. In Fig. 14, the blue circles mark the only designs to satisfy all constraints while in Fig. 15 these globally feasible designs are enclosed within a convex hull.

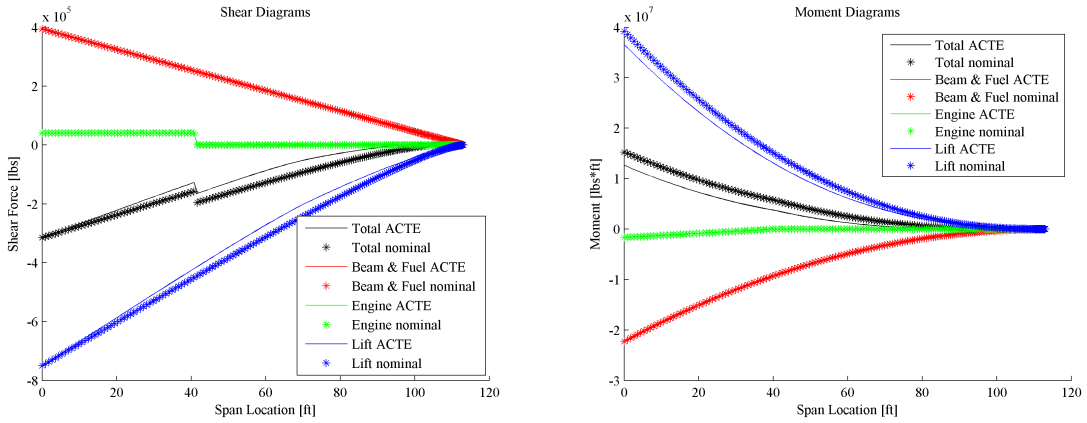

Figure 12: Example Shear Force and Moment Diagrams Plotted against the Wing Semi-span

3.2.2 Conclusions and Consequences

The results obtained in the canonical example problem illustrated the presence of constraints and variable correlations responsible for the boundaries of the feasible space within the hypercubic volume defined by design variable limits. Thus, the design

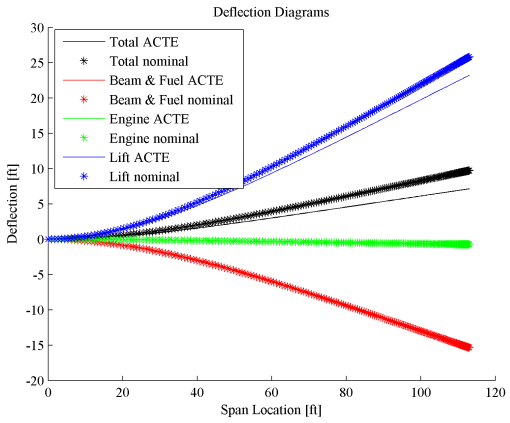

Figure 13: Example Deflection Diagram Plotted against the Wing Semi-span

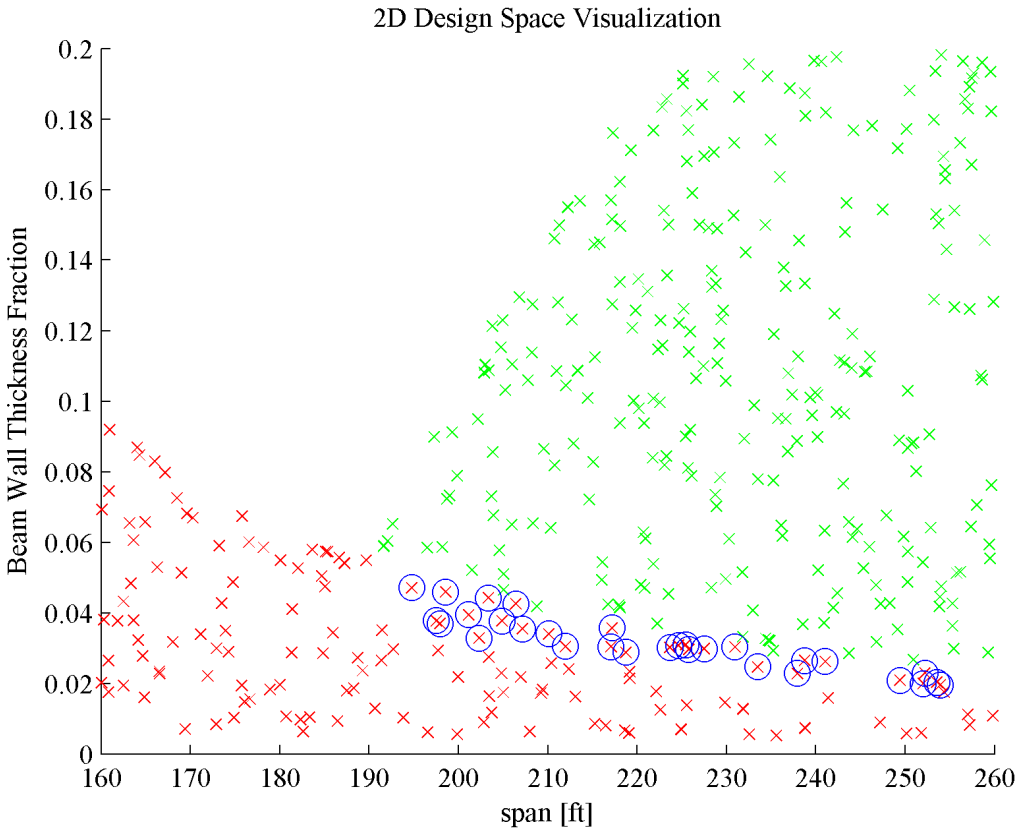

Figure 14: 2D Depiction of the 5D Design Space

space for this problem and by association that of larger aircraft conceptual design problems are assumed to be potentially non-hypercubic. This result also validates

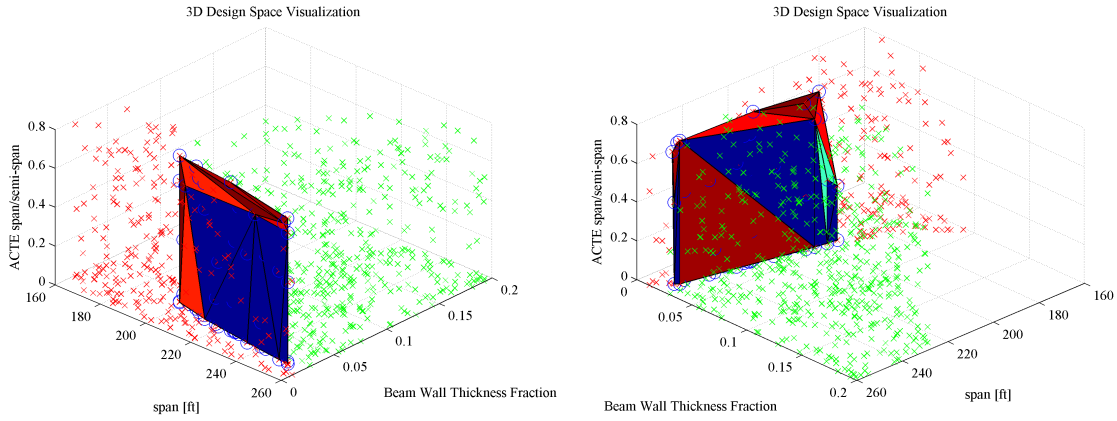

Figure 15: 3D Depictions of the Design Space

the search for a more efficient methodology to explore this class of design space. It can clearly be seen in Fig. 15 from the convex hull used to bound the feasible design cases which exist at the intersection of all constraint defined feasible spaces, that the global feasible design space occupies only a small and irregular volume within the considered hypercubic design space. Thus by seeking out and bounding this non-hypercubic space and then leveraging this representation to guide future design space exploration, it is evident that computational resources can be used with much improved success and therefore efficiency.

CHAPTER IV

METHODOLOGY

4.1 Design Space Exploration Decision Support Methodology (DSE-DSM)

A new methodology is proposed to address the shortcomings of current methods when applied to conceptual design problems which are computationally expensive, non-hypercubic and require repeated exploration. In general, a methodology is sought to provide Design Space Exploration Decision Support for generic spaces of interest recognizing that the above factors, if present, should influence how this exploration is performed. The *top down decision support process* (Fig. 16 - adapted from the Georgia Tech Generic IPPD Methodology) can provide a framework to help identify the pieces of such a methodology [4]. The first step in this process is to 'Establish the Need' which for design space exploration encompasses end goals such as regression/surrogate generation, optimization, visualization and knowledge gain. Current practices can successfully enable these goals for certain types of problems. However, by making the assumption that all design spaces are hypercubic in nature and ignoring important characteristics of the design problem, resources may be used inefficiently and incorrect conclusions drawn due to extrapolating models. Thus it is imperative that this new methodology consider as **input the characteristics of the design space** of interest as well as the availability of the resources used to explore it. Secondly to 'Define the Problem' the unknown properties of the design space must be elicited which may only happen through a sampling of the space. Without expending all available resources, an **initial sample** can allow for **hypercubic classification** of the design space to determine if any additional steps should be taken to depart from

the common practices of exploration using hypercubic sampling methods. If the feasible space is found to be non-hypercubic, additional information must be considered to 'Establish the Value' of a non-traditional approach for design space exploration. The consequences of sampling the space Business As Usual BAU must be quantified as well as the potential benefits of an 'Alternative' approach. Leveraging this information can allow for **informed design space exploration guidance** to help select the best course of action considering the unique problem characteristics. Ultimately a 'Decision' must be made whether the traditional DSE approaches should be abandoned in favor of one which leverages known characteristics of the design space. For problems which are computationally expensive, non-hypercubic and require repeated exploration, **Set-Based Bounded Adaptive Sampling** will likely prove an enabler for the efficient and effective exploration of the design space in question. These steps are combined to form the Design Space Exploration Decision Support Methodology (DSE-DSM) featured in fig. 17. This methodology serves as the Overarching Thesis of this work and as such was tested element wise and as a whole with the experiments performed.

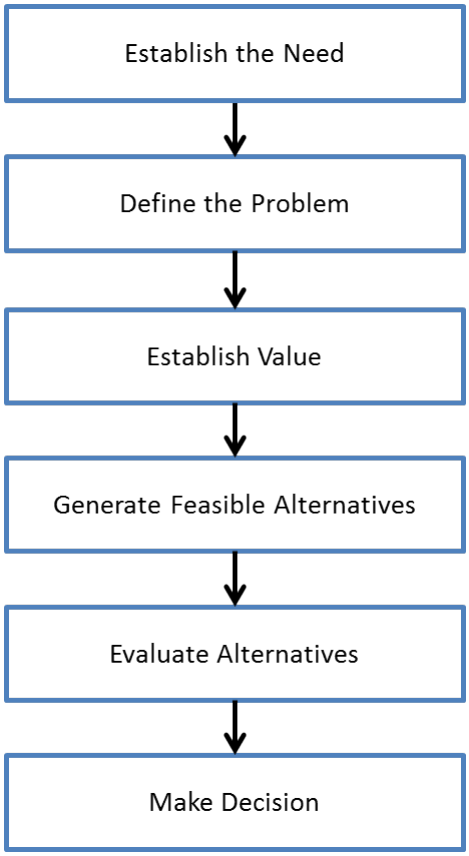

Figure 16: Top Down Decision Support Process Adapted from the Georgia Tech Generic IPPD Methodology [4]

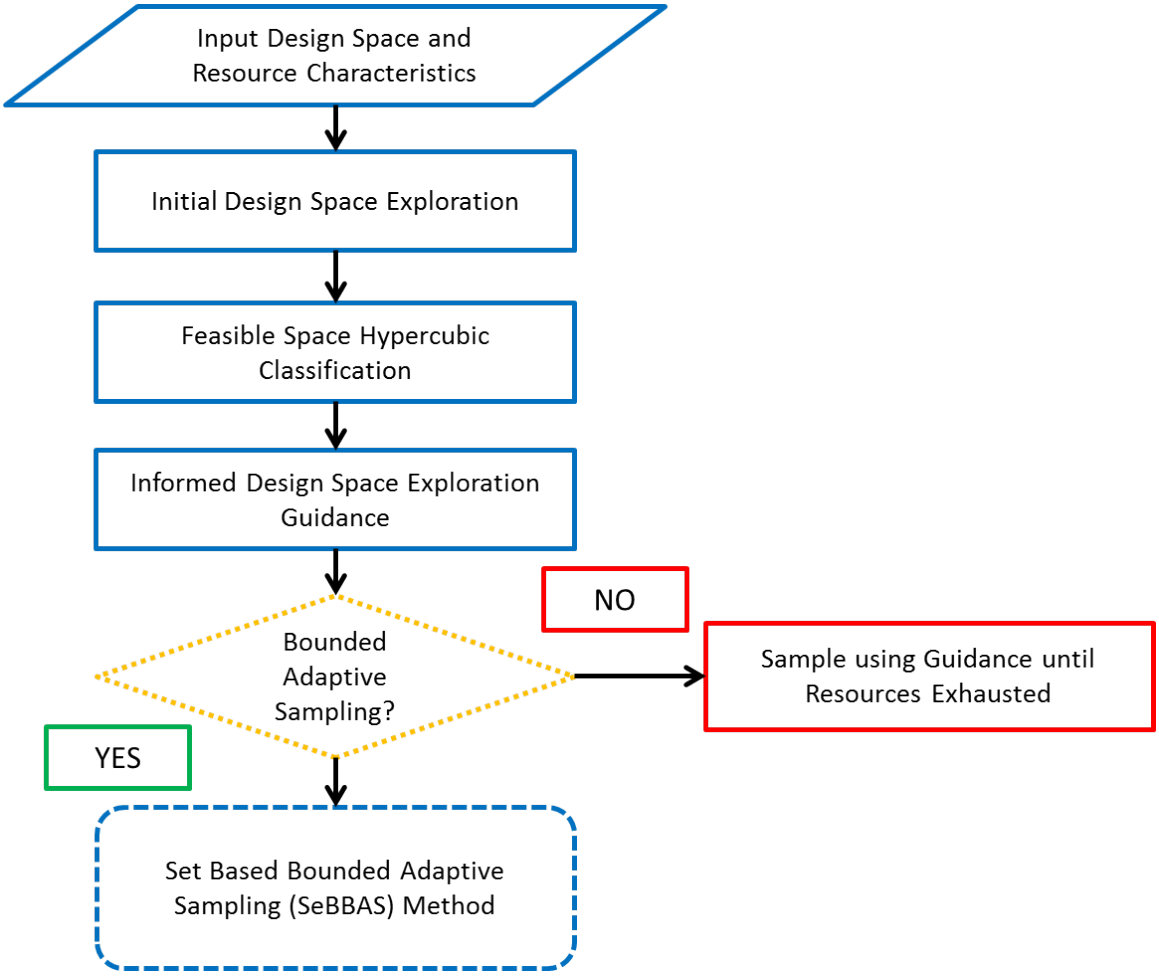

Figure 17: Design Space Exploration Decision Support Methodology (DSE-DSM)

4.1.1 Input Design Space and Resource Characteristics

The first step of the DSE-DSM is to process as input the characteristics of the design space and the available resources. With regard to the design space, attributes of interest are as follows:

- Number of design variables/dimensions (d)
- Design variable limits (this defines the hypercubic design space)
- Design variable type, continuous or discrete (methodology has only been tested for continuous DV)
- Number of constraints or failure modes known a-priori (this helps determine if a Set-Based approach would likely be beneficial)
- Maximum sample budget (n_{max})

The combination of n_{max} and d help determine the ultimate resolution at which the given design space can be sampled and provide guidance for how much of the total sample budget should be utilized for the initial design space exploration. If the design space can only be sampled with a relatively low resolution, then more of the total sampling budget should be utilized for the initial exploration in order to better capture potential features which may make the feasible space non-hypercubic. Conversely, if relatively high resolution can be utilized to sample the space then a smaller percentage of the budget should be utilized for the initial sample (once a critical resolution is achieved) and the rest reserved for adaptive sampling and (potentially) bounding refinement.

The next set of information gathered in this step is utilized to provide an estimate of the relative expense and consequence associated with design space exploration of this particular space. With regard to these subjective metrics, attributes of interest are as follows:

- Characteristics of the Experimental Apparatus (computational or physical, run in serial or parallelized, resource intensive or cheap, setup difficulty)
- Characteristics of the results (can infeasible cases/failures be readily identified, time sensitive)
- Goals: the end use for the information gained (Regressions/surrogates, optimization, visualization, exhaustive search, debugging, etc.)
- Expectations for repeat exploration (will this particular design space or subsets of it be revisited, if so how frequently)

These problem attributes help describe the potential expense and consequence associated for performing design space exploration for the problem under consideration. Due to some characteristics, (ex. expensive physical testing with difficult set-up) adaptive sampling may simply be infeasible. Furthermore, if results are not easily classified, it matters not if the space is non-hypercubic in truth if the output is incapable of revealing it. The end use of the information illuminates the potential consequences of sampling a non-hypercubic space using hypercubic techniques with issues ranging from simple loss of resources to unknown extrapolation occurring within regression models. Lastly, repeat exploration serves a multiplier for expense and makes knowledge about the design space of interest increasingly valuable. Ultimately, the expense and consequence associated with sampling BAU vs. utilizing the DSE-DSM is up to the user to decide, but using the information collected within this step and the next two, the methodology provides guidance to how one should likely proceed under a certain set of likely scenarios.

4.1.2 Initial Design Space Exploration

To perform the initial design space exploration a DOE was desired that maximizes the information return yet was robust to the unknown characteristics of potential features present within a given design space. As non-hypercubic features are potentially expected, the statistical benefits which may come from a structured design (like a Minimax LHS) may be diminished or invalidated due to the presence of infeasible regions of unknown shape and size. This problem is further exacerbated if adaptive sampling is to be performed as even designs such as nested Latin Hypercube Samples may lose their careful structure in a non-hypercubic design space that is repeatedly explored. So as not to purposefully bias any regions of the design space the DOE of choice should attempt to approximate a uniform distribution in all design variables. For all these reasons as well as their heavy use in contemporary problems involving computer experiments, PMC, Quasi-MC and LHS DOE are thus preferred for the initial design space exploration [29].

Once the initial design space exploration has been performed by evaluating the initial DOE with the experimental apparatus for the problem, the output data must be classified. Based upon the experimental apparatus characteristics and the quality/granularity of the output data this classification can be done with differing levels of of resolution. It is desired to be able to bin this date into multiple sets corresponding to individual constraints or mechanisms of infeasibility. For each set available for identification, it is also important to compute the percentage of successful designs, with respect to the total initial sample set, which remain feasible. Ultimately a global feasible set must at least be identified (defined as the intersection of the feasible sets for all relevant constraints) from which to draw conclusions about the nature of the design space through hypercubic classification.

4.1.3 Feasible Space Hypercubic Classification

With the initial sample run and classified, the next step in the DSE-DSM is to classify the feasible design space as hypercubic or non-hypercubic. As highlighted in Observations 2.1.1 and 2.1.2, should specific constraints or significant variable correlation exist within the design space, the feasible space would be made non-hypercubic and thus detection of these features could allow for hypercubic classification. There exist many methods which can detect or estimate specific correlations between variables such as Pearson's correlation coefficient, however a method is desired to capture any relationship linear or non-linear in any number of dimensions existing between design variables. As detailed earlier in this work, Mutual Information appears to fit this requirement perfectly and with such a potential enabler, hypothesis 1 is formally stated:

> **Hypothesis 1: Regarding RQ1**:
>
> If Mutual information is used as a classifier, then for a given design space if the MI value computed for the feasible region is greater than the MI value computed over the entire region sampled, then the Design Space is Non-Hypercubic.

This hypothesis will be tested through experimentation, but assuming for the moment that it can be substantiated, a process is needed through which MI can be calculated for the design space of interest and ultimately a decision made as to the classification of the space. It is important to note that for a general design space, no mater how it was sampled, that the initial design space sample will not be perfectly uniformly distributed in all design variables (even if hypercubic). This is due to the discrete and finite nature of the DOE used to explore this space. Thus a baseline value of MI should be computed for this entire sample with which to ultimately compare to that attained from the feasible subset. It is also important to note that

this baseline MI value (or values computed from drawing multiple repetitions) should strive to contain the same number of cases and thus maintain the same resolution as the feasible sample so as not to unfairly bias the baseline MI calculation. With these considerations observed, the procedure devised by this work for hypercubic classification utilizing Mutual Information is as follows:

1. Generate multiple random samples of a fraction of the initial data set (where this fraction corresponds to the same fraction of design cases that were classified within the set as globally feasible)

2. Compute the MI of these representative baseline samples

3. Compute the MI of the feasible design space (using only the globally feasible design cases)

4. Compute the difference between the feasible space MI value and the baseline MI values

5. classify this space as NHC if the mean of this MI difference (feasible - baseline) accounting for the standard error about the mean is consistently positive and the resolution with which the space has been explored is believed sufficient

The classification results yielded by this element of the methodology provide the final piece of information ultimately utilized to provide informed DSE guidance. Such information provides vital evidence of whether or not abandoning BAU hypercubic sampling techniques could be beneficial or even necessary for exploring the design space in question.

4.1.4 Informed Design Space Exploration Guidance

Following the first three steps of the methodology, important information has been gathered about the design space, the resources available to sample it and the motivation behind performing design space exploration for the problem of interest. For the

purpose of providing guidance of how the space should be explored beyond the initial sample, this information is condensed into three metrics (which were estimated in the previous steps):

- **Tolerable Percentage of Global Infeasible/Failed Designs** TPF**:** measure of how much potential volume is affected by features within the design space and the resources that will be lost if sampling BAU is continued and whether this percentage is tolerable to the user
- **Hypercubic Classification** HC**:** whether the space was found to be hypercubic or not, and to what degree
- **High Expense/Consequence** HEC**:** subjective measure of how expensive it is to explore the design space and the consequence(s) of making incorrect assumptions about the feasible design space

Figure 18 illustrates the decision hierarchy which leverages these metrics to provide informed design space guidance. While not exhaustive, eight probable scenarios are illustrated through the branching of the flowchart each terminating with design space exploration guidance unique to the particular characteristics of the design sapce and problem in question. The Design Space Guidance DSG options presented are elaborated as follows:

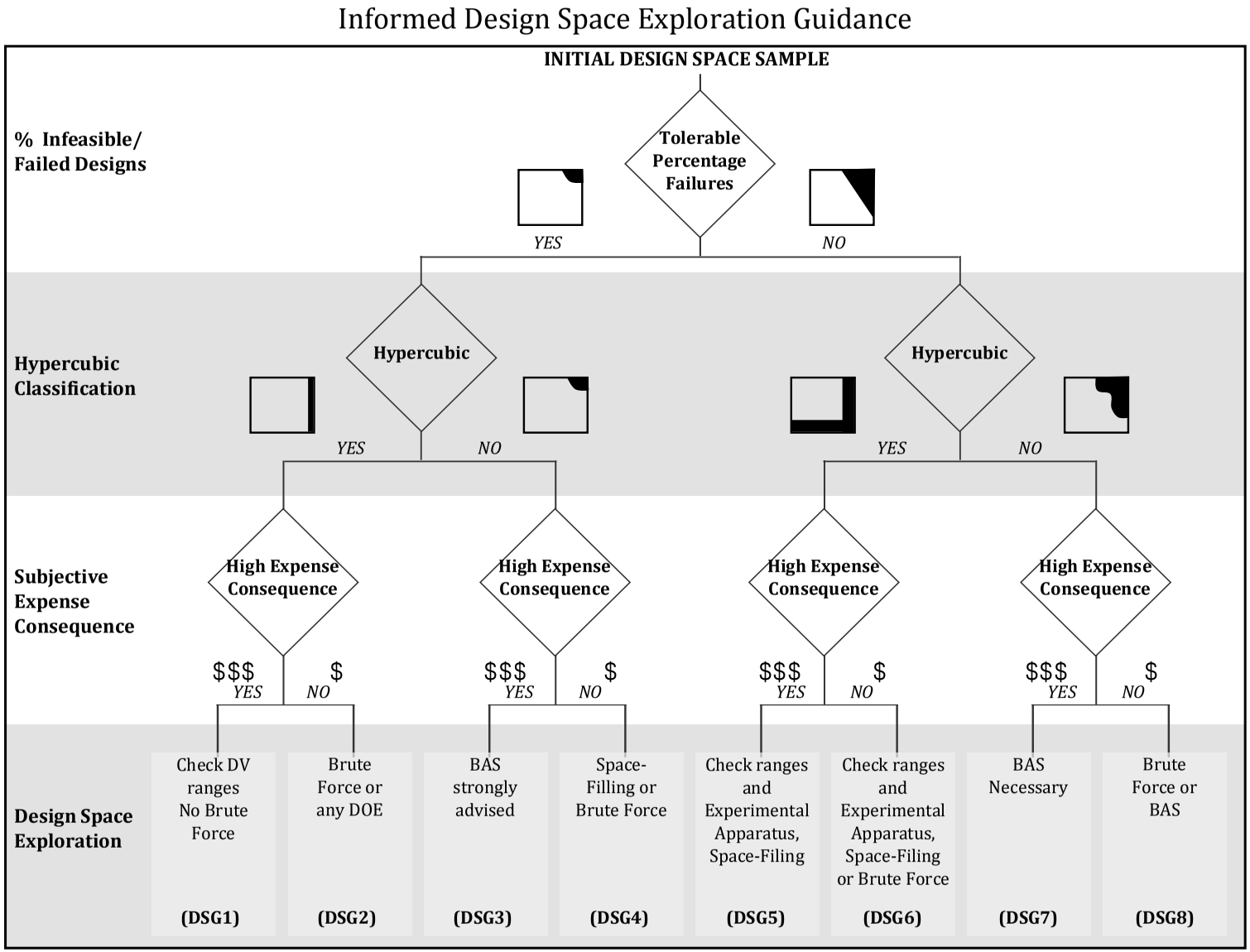

Figure 18: Decision Hierarchy for Informed Design Space Exploration Guidance

1. **DSG1:** (TPF: YES, HC: YES, HEC: YES) Check the ranges on the Design Variables (DV) to ensure no one variable is globally infeasible over some portion of its range. Once this check has been performed, sampling with any DOE except a brute force approach is acceptable.

2. **DSG2:** (TPF: YES, HC: YES, HEC: NO) Continue sampling BAU, all DOE including brute force are acceptable, as failures are tolerable and expense is low it is unnecessary to check ranges.

3. **DSG3:** (TPF: YES, HC: NO, HEC: YES) Bounded Adaptive Sampling (BAS) is strongly advised, can continue sampling BAU but beware risk of extrapolation and the potential presence of correlations/constraints within the design space.

4. **DSG4:** (TPF: YES, HC: NO, HEC: NO) Space-filling DOEs are recommended, brute force may be acceptable if available, BAS is advisable but not necessary (may want to at least identify NHC variables).

5. **DSG5:** (TPF: NO, HC: YES, HEC: YES) No underlying structure was found in the design space, failures appear random or due to aggressive DV range selection, check DV ranges and for convergence or numerical issues if possible. Failures may be arising from internal optimizer with ill-conditioned initial guesses. Unknown variables not explicitly expressed in DOE may also be at play. After experimental apparatus has been examined, continue sampling with a space-filling DOE. Bounding is likely unhelpful or inconclusive until greater sampling density is achieved within the design space.

6. **DSG6:** (TPF: NO, HC: YES, HEC: NO) No underlying structure was found in the design space, failures appear random or due to aggressive DV range selection, check DV ranges and for convergence or numerical issues if possible. Failures may be arising from internal optimizer with ill-conditioned initial

guesses. Unknown variables not explicitly expressed in DOE may also be at play. After experimental apparatus has been examined, continue sampling with a space-filling DOE or brute force if possible. Bounding likely unhelpful or inconclusive until greater sampling density is achieved within the design space.

7. **DSG7:** (TPF: NO, HC: NO, HEC: YES) BAS is necessary, allow for multiple iterations of BAS for greatest effect. Exercise caution when fitting regressions to this space and utilize BAS results to monitor their applicability.

8. **DSG8:** (TPF: NO, HC: NO, HEC: NO) BAS is strongly advised particularly if the space will be repeatedly explored, allow for multiple iterations of BAS for greatest effect. Can continue sampling BAU but will still experience unacceptable percentages of infeasible/failed designs. Exercise caution when fitting regressions to this space and utilize BAS results to monitor their applicability.

This design space exploration guidance provides a path forward for future exploration of the design space informed by qualities observed from initial sampling and analysis. Many paths forward advocate to continue sampling BAU, and for these paths, here is where the DSE-DSM ends. However, for those scenarios in which BAS is advised the methodology transitions into its final (but considerable) step in which Set-Based Bounded Adaptive Sampling is performed to more efficiently and effectively explore the design space of interest.

4.2 Set-Based Bounded Adaptive Sampling (SeBBAS)

With the decision made to pursue a path of adaptive sampling in order to leverage information previously generated about the design space yet conserve future experimental effort, the question arises of how this should be performed. Given that the design space is likely non-hypercubic (as this path was chosen) the challenge is now to

efficiently sample a design space with unknown interior boundaries and regions of feasibility. Research question 2 suggested a solution to this and asked if the construction of a bounding for this non-hypercubic space would be generally useful for its future exploration. While seemingly trivial, other factors at play such as the resolution of the design space and potentially the bounding method used could make this question not so easy to definitively answer. Therefore, hypothesis 2 is formulated to explore this question through experimentation and determine if adaptive sampling leveraging a bounding is a useful path forward.

Hypothesis 2: Regarding RQ2:

If a bounding is constructed using sufficient resolution to resolve the features present within a non-hypercubic design space, then it can be leveraged to enable more resource efficient future exploration of the space.

Assuming again that hypothesis 2 can be substantiated the elements required for a bounded adaptive sampling method are elaborated. The computational expense of problems requiring adaptive sampling necessitates intelligent **computational resource management** and conservation whenever possible with great emphasis placed on the elimination of waste resulting from the evaluation of designs in infeasible regions. This requirement brings to light the importance of determining the boundaries of the feasible design space. Because of the design problem characteristics, it is likely subject to many constraints of various types as well as design variables with potentially significant correlations. These relationships may or may not be known a-priori, and the design space must be sampled thoroughly to determine where these limiting features of the design space exist. **Adaptive sample generation** methods will prove useful in this endeavor by using guidance from discoveries made in previous sampling iterations to steer future samples away from regions denied by constraints and/or variable correlations. Assuming this design space exists for a computational

problem, as design points are selected to be evaluated throughout the design space they must be evaluated using a **physics-based computational analysis** which captures all of the relevant physics, assumptions and requirements for the design problem being explored. From this analysis, the performance of each design can be determined along with its standing with respect to all of the constraints. At this step in the process it is necessary to provide a **classification of results into sets**, binning designs in order to enable the mapping of the design space and ultimately inform the next iteration of designs to be considered. Some designs may be globally feasible and satisfy all constraints, while others may only satisfy some constraints and others still none at all. Some designs may not even return valid output, due to computational model limitations such as convergence failure or exceedance of model applicability ranges. Once classified, these designs can now be utilized to construct a feasible **Machine Learning based design space bounding**. Leveraging the ideas of Set-Based design, sets can be formed for individual constraints and regions of partial feasibility and then 'bound' using Machine Learning classificiation algorithms. Determining the intersection of these bounded sets will yield an estimate for the true feasible design space [48]. With this estimate, new sample points can be suggested which attempt to expand or refine this mapping or simply more thoroughly explore the already identified feasible design space. Ultimately, through iterative adaptive sampling guided by this set-based bounding of the design space, computational waste can be progressively eliminated and an accurate and well sampled representation of the feasible design space produced. This product can then be used to generate regression models, perform bounded optimization, visualize the design space or simply allow more efficient future DSE. Eventually, when the problem must be revisited with updated constraints and/or assumptions, the bounding will serve as an excellent starting guide for intelligently and efficiently probing the updated problem.

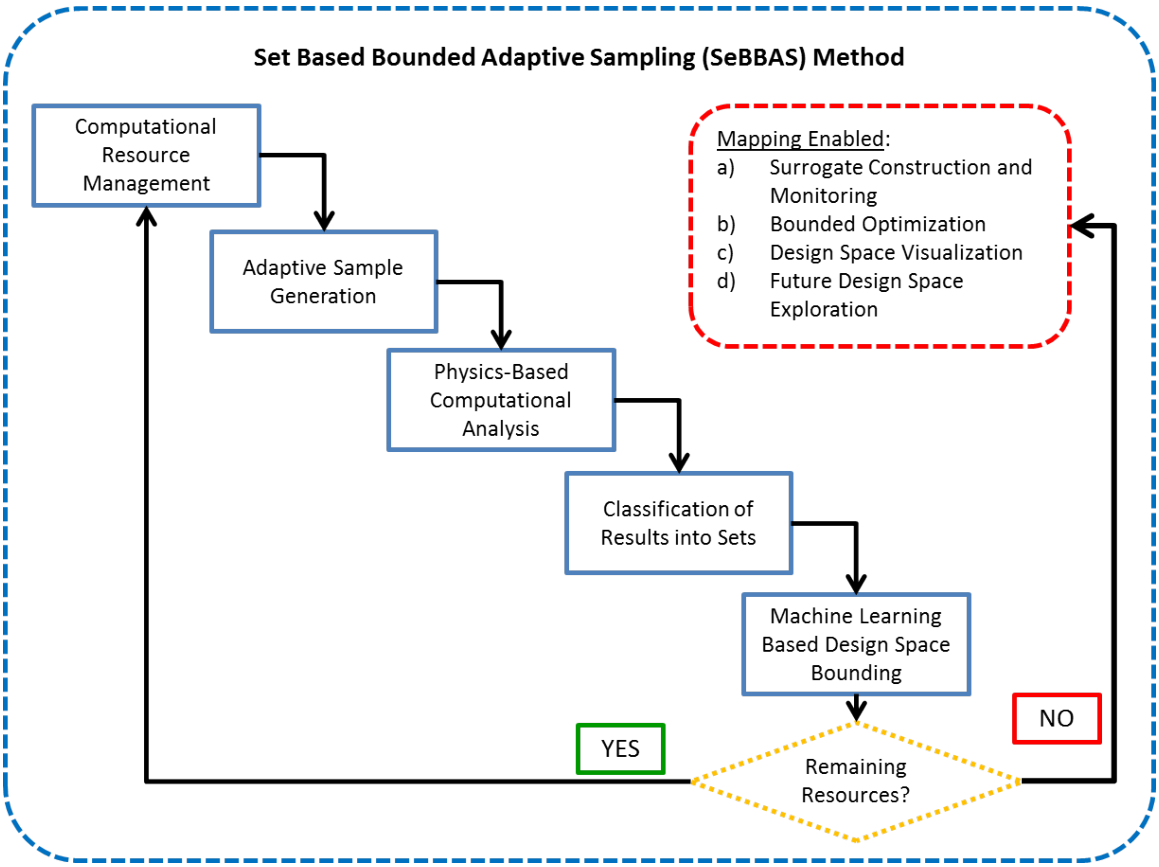

Figure 19: Set-Based Bounded Adaptive Sampling (SeBBAS) Method

4.2.1 Computational Resource Management

This element of the methodology is responsible for tracking and allocating computational resources. Because the methodology is based on an iterative sampling and mapping of the design space, this element is necessary to ensure that successive iterations have sufficient computational resources to exploit the discoveries of the previous iterations. For this reason it is desirable to hold a percentage of the available computational resources in reserve for future iterations. However, if the resource management is too stingy with resources during the onset of adaptive sampling, the bounding produced from initial design space exploration of the feasible design may not be refined enough to provide an accurate representation of the space or even identify all possible

regions of feasibility. However, excessive computational expenditure in the initial iterations may waste computational resources by over-exploring infeasible regions (where the bounding is ill-defined or inaccurate) and leaving too few cases left to refine and thoroughly explore feasible areas. This time dependent placement of samples is very similar to objectives of stochastic optimization methods like genetic algorithms and simulated annealing. In both cases, it is generally desired to first explore the entire design space then gradually transition to exploitation of promising regions. Such an approach toward resource management is utilized for this method with the particular DSE end goals used to determine whether emphasis is placed on refining the NHC boundaries or sampling the bound feasible space.

4.2.2 Adaptive Sample Generation

The adaptive sample generation element is responsible for determining the placement of designs to be evaluated for the current iteration. As the boundings are constructed for the individually classified feasible sets, this information will be utilized to guide the sample sets of the following iterations. Based on the current priorities of the resource management element, candidate designs can be suggested which have a high likelihood of refining the feasible design space boundary, improving sample density in sparsely sampled regions or simply exploring the currently defined feasible space. Candidate designs are ranked and then selected using a threshold or quota limit for the current iteration. This is achieved through querying the bounding classifiers about points that contain the greatest classification uncertainty (these become suggested designs for boundary refinement) or have low classification uncertainty and are deemed feasible (these become suggested designs exploitation and feasible space exploration). Those candidate designs deemed likely infeasible by the classifiers can be used for exploration and to add resolution to regions currently believed infeasible. This process of adaptive sampling will continue until all computational resources have been expended.

4.2.3 Physics-Based Computational Analysis

The physics-based computational analysis element is where the evaluation of design alternatives occurs. Once the designs have been selected by the sampling method, the cases can be run in this environment to determine their performance and adherence to constraints. In general, for the larger class of problems described simply as computationally expensive, non-hypercubic and requiring repeated exploration, this method could contain any analysis which can transform sample point inputs into responses.

For the specific design problem considered in this dissertation, this element is composed of a state-of-the-art multidiciplinary physics-based conceptual design environment (EDS) in which advanced aircraft concepts with the infusion of emerging technologies can be evaluated. In order to reduce the computational expense for the purposes of performing experiments with this environment, the aircraft noise modules will be deactivated to decrease individual case run time.

4.2.4 Classification of Results into Sets

In order for designs to be of use in constructing a bounding for the design space they must first be binned into sets of designs with similar characteristics and classified according to what regions of the design space to which they belong. To accomplish this, designs are collected into **Constraint Defined Feasible Sets** $CDFS$ which are defined as follows [48]:

Let H $\subset \mathbb{R}^{\mathrm{d}}$ be the d-dimensional hypercube defining the extent of the design space

s.t. a unique design can be expressed as $X_j = [x_{1j}, x_{2j}, \ldots, x_{dj}] \in H \forall j = 1, 2, \ldots, n$

where $x_{ij} \in [x_{iLowerLimit}, x_{iUpperLimit}] \forall i = 1, 2, \ldots, d$

$$\text{Let } F_k(X_j) = \begin{cases} 1 & \text{if } X_j \text{ is Feasible w.r.t Constraint k} \\ 0 & \text{if } X_j \text{ is Infeasible w.r.t Constraint k} \end{cases}$$

Then the Constraint Defined Feasible Set for the kth constraint is given by:

$$CDFS_k = \{X_j \in H : F_k(X_j) = 1\} \forall j = 1, 2, \ldots, n$$

s.t. for a total of C Constraints, the Global Feasible space $G = \bigcap_{k=1}^{C} CDFS_k$

By collecting the classified designs into sets in this manner, the global feasible design space is inherently given by the intersection of all the CDFS. Such an approach is advantageous because as in Set-Based Design, the boundary of a given CDFS may only be a function of a subset of the total number of design variables. This is potentially valuable because it implies that for a given design space sample budget the effective resolution with which a given CDFS has been sampled may be greater than that with which the global feasible set has been sampled. It is likely that with greater sampling resolution comes more accurate boundings and thus hypothesis 3 is stated:

Hypothesis 3: Regarding RQ3):

If a set-based design (SBD) approach, which integrates through intersection multiple Constraint Defined Feasible Sets (CDFS), is used to construct a global boundary of the feasible design space, then this approach will provide a more efficient and accurate representation of the true feasible space than simply bounding the global feasible set.

4.2.5 Machine Learning Based Design Space Bounding

The bounding of the design space is of critical importance to the effectiveness of the iterative adaptive sampling employed by the SeBBAS method. The bounding must be refined enough to construct a adequate representation of the design space to assist in the identification of areas of infeasibility and possible limitations in the computational environment. However, because the problem considered is also computationally expensive, it is desirable to construct and then thoroughly explore this global bounding with the minimum amount of resources as possible. Key to construction of the global bounding is the determination of boundaries within the design space that exist between designs of different classifications. These boundaries will be defined by

constraints, correlated variables and computational method limitations. Two current state-of-the-art machine learning classification techniques, random forests and kernel-based support vector machines, are utilized to determine these boundaries. These techniques were chosen based upon their general robustness, applicability in high dimension and proven utility for solving many different classes of problems with little known a-priori about the characteristics of the design space [21].

In order to take advantage of the potential for increased effective resolution, alluded to in hypothesis 3 through use of a set-based approach, relevant design variables (those involved in defining the boundary) for each CDFS must be identified. To accomplish this, variable importance rankings provided by random forests bound in all design variables to each CDFS combined with cross validation error minimization will be used for feature selection. These design variable subsets, unique to each CDFS, will then be used as the only features with which the CDFS bounding classifiers will be trained. Then utilizing these set-based boundings, the classification of new candidate designs can be simultaneously predicted by each of the CDFS bounding classifiers to provide estimates for cases which have a high probability of being globally feasible or in the vicinity the boundary. These CDFS bounding classifiers can then be retrained and refined in successive iterations, utilizing the data from the adaptive samples they suggested. Such a bounding procedure will continue until resources are exhausted or satisfaction with the representation of the design space reached.

CHAPTER V

EXPERIMENTAL RESULTS

The research plan for this thesis is focused around testing all stated hypotheses within relevant experimental environments and in doing so, seeks to provide a solution to the overarching research objective. Should this objective be satisfied, the methodology proposed within will enable the efficient exploration of large multidimensional design spaces required for practical aircraft conceptual design problems and provide a representation of these design spaces superior to any which could be attained from existing methods for similar computational effort. To guide the development of the experiments, the overarching research objective is summarized:

Overarching Research Objective Summary:

This thesis seeks a general methodology to provide decision support for design space exploration for general design spaces through:

- Providing Hypercubic Classification
- Constructing Constraint Defined Feasible Sets
- Bounding NHC Feasible Design Spaces
- Improving Efficiency of DSE Resource Use

In order to test such a methodology and whether it can adequately perform the above functions, four experiments were devised to examine critical elements within the methodology and its practical functionality as a whole. Table 4 summarizes these four experiments and the hypotheses which they test.

Table 4: Experiment-Hypothesis Summary and Mapping

Experiment	Hypothesis Test
I	MI is an appropriate Hypercubic Classifier
II	Bounding the NHC feasible design space is useful
III	Set-Based is superior to Global Bounding
IV	Methodology is useful for a practical problem

5.1 Experiment I

5.1.1 Motivation and Thought Experiment

The motivation for this experiment is rooted in the deficiencies highlighted in the initial efforts to provide Hypercubic classification for the design space given by the Canonical Example Problem. It was originally posited that observation of characteristics such as variable correlations or the presence of constraints in the design space would allow for it to be classified as Non-Hypercubic. While these initial positions were shown to be correct, they do not encompass the full set of conditions through which a feasible design space could be made Non-Hypercubic. Furthermore, discovering these attributes this way involved the construction of a design space bounding and estimation of various correlation coefficients. To address these limitations, a more general and less expensive means of Hypercubic classification for a design space was sought. A means of classification was needed that could simultaneously observe correlation (linear and non-linear) between variables, voids, and where constraints (both known and unknown) were making regions of the space infeasible. The Mutual Information (MI) metric has been shown to illuminate various non-uniform features of high dimensional spaces and was thus selected for investigation as a possible Hypercubic classifier [45, 52].

5.1.1.1 *Hypothesis 1: Regarding RQ1*

If Mutual information is used as a classifier, then for a given design space if the MI value computed for the feasible region is greater than the MI value computed over the entire region sampled then the Design Space is Non-Hypercubic.

5.1.2 Experiment Design

In order to verify that MI was an appropriate and robust classifier, multiple design spaces, both Hypercubic and Non-Hypercubic and of differing dimension and sampling resolution were examined. For the hypothesis presented in this experiment to be substantiated, MI must reliably distinguish Non-Hypercubic Spaces from Hypercubic ones. To provide variety in the design spaces examined, five different methods of sampling were used to populate the design spaces and 20 separate constraints were utilized to produce features which could appear in generic design spaces.

5.1.2.1 *Apparatus*

In order to evaluate the potential for MI to be utilized as a Hypercubic Classifier, some representative Hypercubic and NHC test design spaces were required. A series of d-dimensional linear and non-linear constraints were developed which could be applied to a baseline Hypercubic design space sample of interest to either produce a reduced Hypercubic or NHC feasible space defined by the designs which satisfied the given constraint. Each of the constraints were applied numerically (See Appendix for details) to the baseline Hypercubic design spaces defined for each repetition of each unique design space sample (combination of settings from the test matrix).

With the test design spaces generated, this experiment now required the estimation of Mutual Information for the numerous design spaces of differing construction, dimension, resolution and structure. As such a large number of design spaces were to be evaluated, a computational tool for estimating MI for d-dimensional spaces with finite samples was sought. Based upon its success in literature in estimating

MI values in high dimensions from finite sample sets, the Windows version of the Mutual Information Least-dependent Component Analysis $MILCA$ was utilized to compute the MI for each of the unique design spaces [95, 94, 50, 51, 7, 8, 66]. The 'MIhigherDimension' function was utilized with default parameters and a k-nearest neighbors value of 6 for all computations of MI presented within this work. This function works by applying the kth nearest-neighbor binned mutual information estimation algorithm presented in Kraskov et al. over a set of finite samples in multiple variables [50].

5.1.2.2 Metrics

Mutual Information for each unique design space was tracked as the primary metric in this experiment. However, since the number of samples contained within each DOE is non-infinite, a baseline MI value was calculated for each design space before constraints were applied to generate the subset of cases that would compose the feasible space. An MI value was then calculated for this feasible space. To determine if a design space was Non-Hypercubic, the baseline and constrained MI values were combined into a single metric called 'MI Delta' or MID. In addition, in order to track the percentage of the design space made infeasible by the application of individual constraints, a 'successful case percentage' SCP metric was was also computed. This metric when combined with MID could allow for decisions to be made about how to continue further sampling of the design space. Its consideration is important as a space may be deemed NHC by MID alone yet if only a small percentage of cases are actually infeasible, bounded adaptive sampling may not be necessary for future design space exploration. The metrics for Experiment I are defined as follows:

- **Mutual Information Delta** MID: the difference between the baseline MI value for a given unique DOE and the MI value calculated for the cases within that DOE which remained feasible after a given constraint was applied. If the

MI delta value is positive (i.e. the MI value of the constrained space is higher than that of the baseline) then features exist within the design space that make it Non-Hypercubic.

$$MID = MI_{Feasible} - MI_{Baseline} \tag{27}$$

- **Successful Case Percentage** SCP: the percentage of cases with respect to the baseline DOE which remained feasible after a given constraint was applied. This metric can be seen as a measure of the severity of whatever phenomenon is causing cases to become infeasible within the design space.

$$SCP = \frac{FeasibleCases}{TotalCases} \tag{28}$$

5.1.3 Experiment Settings and Execution

This experiment requires both Hypercubic and Non-Hypercubic design spaces to be examined and classified. To be of use as a general classifier, MI must be reliable not only in detecting the various features which can lead a design space to be non-Hypercubic, but also be capable of handling design spaces of differing dimension, resolution and initial sampling methods. For these reasons the following test matrix was constructed:

5.1.3.1 Test Matrix

- **DOE**: Pseudo-Monte Carlo PMC, Sobol Sequence with Matousek-Affine-Owen scrambling $SSMAO$, Latin Hypercube Sampling LHS, Latin Hypercube Sampling Minimax Optimized $LHSMM$, Latin Hypercube Reduced Correlation $LHSRC$ **Note: these particular DOE were chosen to be a representative set of space filling designs which are popular for computer experiments, as such these five designs include two 'random' designs and three Latin-Hypercubes with varying degrees of DOE optimization

- **Dimensions (d)**: 2, 3, 4, 5, 10, 20, 50, 100-PMC Only
- **Cases (n)**: 100, 1000, 10000
- **Constraints**: Hypersphere HS, Hypersphere 2-D $HS2$, Reduced Hypercube Single RHS, Reduced Hypercube Single 1-D $RHS1$, Reduced Hypercube Multiple RHM, Reduced Hypercube Multiple 1-D $RHM1$, Random Removal Fixed Percentage $RRFP$, Random Removal n/d $RRND$, CheckerBoard Coarse CBC, CheckerBoard Coarse 2-D $CBC2$, CheckerBoard Fine CBF, CheckerBoard Fine 2-D $CBF2$, Linear Constraint Small LCS, Linear Constraint Small 2-D $LCS2$, Linear Constraint Large LCL, Linear Constraint Large 2-D $LCL2$, Non-Linear Constraint Small $NLCS$, Non-Linear Constraint Small 2-D $NLCS2$, Non-Linear Constraint Large $NLCL$, Non-Linear Constraint Large 2-D $NLCL2$ **See Appendix for Details**

5.1.3.2 Procedure

The following steps describe the procedure through which MI was calculated for each of the combinations in the test matrix and how classification judgments were made:

1. Select from the test matrix a unique combination of DOE type, number of dimensions and number of cases - this defines a design space
2. Generate 60 repetitions of this design space using the sampling method prescribed by the DOE type and save these repetitions
3. For each repetition, save a baseline in which no constraints are applied to the design space (i.e. no points are classified as feasible or infeasible and are all kept)
4. For each repetition, individually apply each constraint and save the corresponding feasible design spaces which result

5. For each repetition, compute the MI value for the baseline

6. For each repetition, compute the MI value for all of the constrained design spaces

7. For each repetition, compute MID between each of the constrained design spaces and the baseline design space

8. Compute the mean and standard error of the MID over all repetitions

9. Draw conclusions for that particular type of design space

   (a) If the mean MID value > 0 and the lower error bound is $>= 0$, then the design space is classified as NHC

   (b) If the mean MID value is $<= 0$ then the design space is classified as Hypercubic

   (c) If the mean MID value > 0 and the lower error bound is < 0, then the MI classifier test is inconclusive and cannot determine whether the space is Hypercubic or NHC, more resolution is likely needed

10. Repeat steps 1-9 for all unique combinations within the test matrix

11. Draw final conclusions

5.1.4 Results Discussion

Due to the large size of the test matrix evaluated within this experiment which amounted to over 2500 unique calculations of MID and SCP, a suitable combination of variables was sought to collapse the output data against and thus allow general conclusions to be drawn. Initial analysis of the raw data set illustrated an intuitive but perhaps not obvious result; the success of MI as a Hypercubic Classifier was dependent upon the resolution at which a given design space was sampled. This makes

sense as MI is essentially being used, by examining the distributions of the design variables, to discover or resolve features and structures which exist within the design space and potentially make it NHC. It follows then that if a design space is 'sparsely' sampled then these features, should they exist, would be harder to resolve and thus produce widely varying MID values between repetitions. This trend was roughly observable in the data as MID values resulting from the application of a particular constraint (meant to make the design space NHC) grew larger and more closely distributed between repetitions with an increasing number of cases and decreasing number of dimensions defining a unique design space. From this it was concluded that some measure of resolution would likely serve as a good similarity parameter for the data, but it was not apparent how this 'resolution' should be represented.

Literature regarding design of computer experiments often suggests the use of n/d as a measure of sampling resolution to ensure enough cases (n) are allocated to each dimension (d) to accurately capture behaviors within a design space [55]. This representation of design space resolution was initially tested on the MID output data, but produced unsatisfactory results. The data showed that design spaces under the application of the same constraint and with the same n/d values would differ greatly in MID values (see Fig. 20). From these observations it appeared that the loss of resolution due to an increase in dimensionality for a given constrained design space was not made up for by increasing the number of cases required to maintain the n/d ratio. When explored further, this makes sense as n/d is a criterion often used to ensure there is enough resolution within the design space to produce accurate regressions of responses. As a Hypercubic Classifier, MI is meant to resolve features within the design space, not the response space and thus suffers more from the curse of dimensionality as the number of design variables increases. This result suggests that perhaps an appropriate resolution metric for MI as a Hypercubic Classifier of a given design space should vary exponentially rather than linearly with the number of

dimensions of the design space in question.

Given the hypothesis that the appropriate resolution metric should likely vary exponentially with the number of dimensions (d), a concept for design space sampling resolution was explored which drew inspiration from the construction of Full Factorial DOE. In the creation of Full Factorial DOE, the number of cases required for the DOE is defined by the number of levels or settings represented in each design variable and the number of design variables considered. The number of cases required is computed as follows:

$$Cases_{Required} = (FactorLevels)^{Dimensions} -> n = L^d \tag{29}$$

Using this formula, for example, a full factorial DOE for 3 design variables with 2 levels each would require 8 cases. This is another way of saying that with these 8 cases, a design space spanned by 3 variables can be resolved at 2 levels per dimension. While it is necessary to have an integer number of cases and dimensions, it is not necessarily required to restrict the level of resolution to integer values only. With this realization, a new resolution concept and similarity parameter, coined 'Levels Per Dimension' or LPD was devised in order to provide a continuous metric for design space sample resolution, generalized to non-integer values, to represent the the number of equivalent factor (design variable) levels present within a design space. LPD is at defined mathematically as follows:

$$n = L^d -> LPD = n^{\frac{1}{d}} \tag{30}$$

To illustrate how LPD differs from n/d as a resolution metric we can look at the example of a 2-dimensional design space explored by 16 cases. In this example, the value of n/d is 8 and thus were the design space extended to a 3rd dimension, to sample this new space with the same resolution under this metric would require the addition of another 8 cases for a total of 24. The LPD for the original design space however is equal to 4 and if the design space were similarly extended to a 3rd

dimension, in order to maintain the same resolution using LPD as a metric would require the addition of 48 cases for a total of 64.

Utilizing LPD as a resolution measure better reflects the exponential nature of design space volume growth associated with the extension of the design space to additional dimensions. Fig. 20 illustrates this as MID is plotted against both n/d and LPD for two NHC feasible design spaces. In both methods sufficient resolution is required before the MID values for all DOEs are consistently positive (and thus indicate the design space is NHC). However, the trend is much more clearly visible using LPD as the similarity parameter for the data (sampled at multiple different combinations of DOE type, number of cases and dimensions) collapse more cohesively. Most importantly, with n/d as the similarity parameter, it is difficult to identify a critical value of 'resolution' after which the structure of the design space is correctly and consistently classified by MID. Conversely, LPD provides a much clearer picture of the resolution required to resolve the particular features of these NHC spaces as after a certain value the data consistently take on a value of MID greater than zero and ultimately converging to a particular value.

Using LPD as a resolution metric provided a much more appropriate means to view MID results across the entire test matrix and allowed general conclusions to be drawn across the entire output data set. For the remainder of this investigation, LPD was utilized as the metric for resolution and was integral in determining the requirements for successful implementation of elements of the methodology. The following figures illustrate the results of Experiment I viewed through the metric MID as a function of LPD. It is important to note that the LPD values presented represent original LPD values where the n value used to compute LPD is the number of cases in the baseline unconstrained design. This convention is used as one would not know a-priori how many cases would be denied or made infeasible by unknown features existing within the design space.

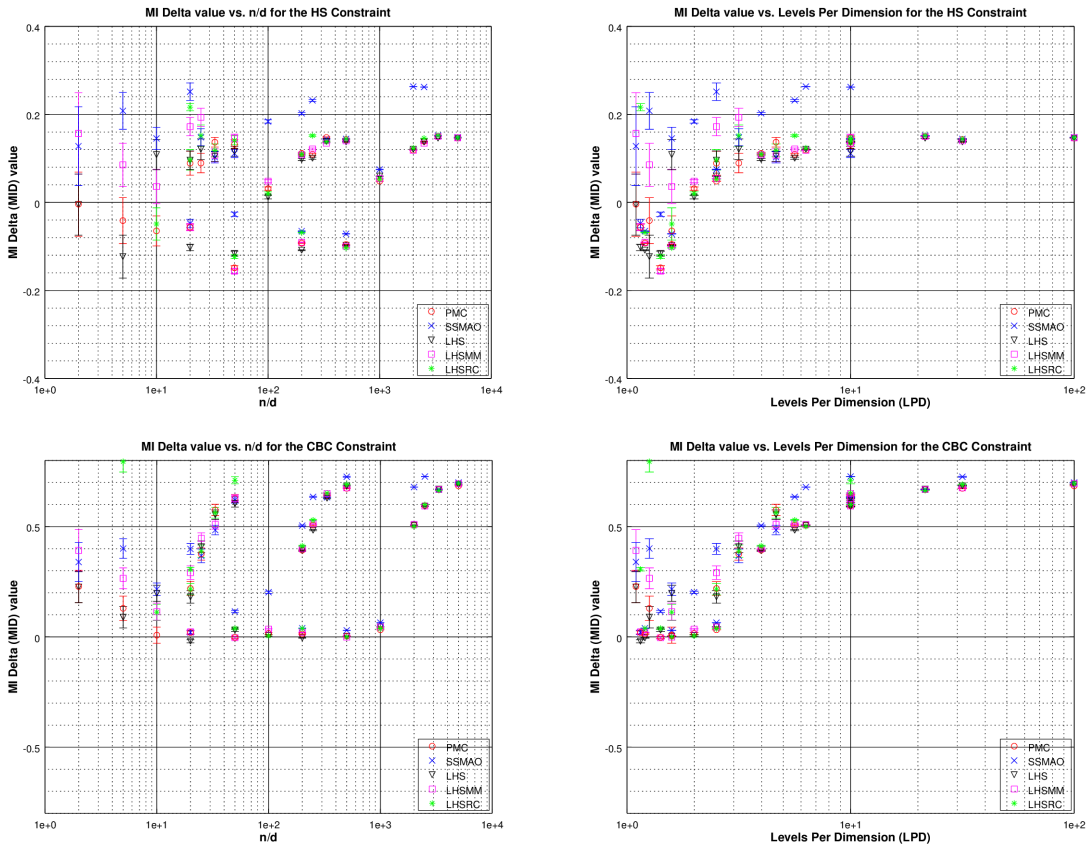

Figure 20: Comparison of n/d and LPD as similarity parameters for the Hypersphere (HS) and Checkerboard Coarse (CBC) Constraints

Results Interpretation (Figs. 21-40) The following figures illustrate the results of utilizing MI as a Hypercubic Classifier for all of the unique design spaces specified in the test matrix. Each separate figure highlights the results for one particular type of constrained design space described in the figure's title. The resolution similarity parameter LPD is plotted on a log scale on the x-axis, with resolution increasing as LPD increases. MID is featured on the y-axis with a linear scale and indicates the difference in MI values computed between the baseline unperturbed Hypercubic design space and the design space which resulted with the application of the particular constraint. Five different marker types are used to illustrate the results of the different DOE types tested. The placement of each marker signifies the mean of the MID results over the 60 replicates examined at a given LPD value while the error bars

span the standard error about the mean in both directions. A solid red line is plotted through the PMC data to show how the results behave as the resolution (LPD) is increased. If the MID values are negative or approximately zero, then the given design space is classified as Hypercubic. Positive MID values however indicate a NHC classification. A thick vertical red line is used to indicate the critical LPD value (tied to the PMC data) after which MI is consistently able to provide the correct classification (respecting the error about the mean) for the constrained design space being evaluated. Generally at low sample resolutions (LPD) there is significant noise in the data and the classifications are unreliable, but as LPD increases, the data ultimately converge on MID values which provide the correct classification for the given space.

In order to ensure that MI would not classify Hypercubic design spaces as NHC and thus provide false positives, the first set of constrained design spaces examined were Hypercubic. The Reduced Hypercube constrained design spaces (RHS, RHS1, RHM, RHM1) created infeasible regions within the original design space in such a way that the remaining feasible space was simply a single hypercube or multiple hypercubes of smaller hypervolume. Figs. 21 through 24 illustrate the results of the MID classification tests performed on these spaces. Perhaps the first notable result is that there does appear to be an effect due to different DOE types. In general, it seems that the more structure built into the DOE, the more it is affected in terms of MID with the removal of designs. Because of this perturbation of their original structure, designs such as SSMAO, LHSMM and LHSRC appear to require more resolution (higher LPD) to return a MID value near zero and thus properly classify the design spaces as Hypercubic. Additionally, at low values of LPD (roughly those less than 2), MID estimates fluctuate significantly between repetitions and do not provide reliable classification of the spaces. Ultimately however, given enough resolution, MI is able to consistently classify the design spaces as Hypercubic for all DOE types.

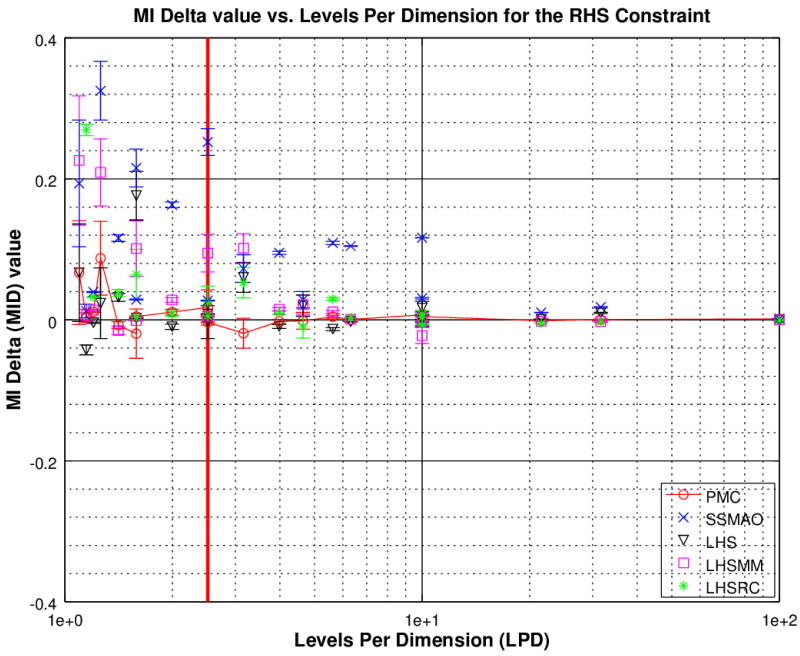

Figure 21: MI Classification Results for Reduced Hypercubic Singular (RHS) Design Spaces

When comparing the differences between the MID results for the design spaces which were subject to the constraints in all dimensions (Figs. 21 and 23) against those that only had the constraints applied to a singular dimension (Figs. 22 and 24) it appears that the spaces subjected to constraints in only one dimension were slightly easier to resolve and could be classified as Hypercubic at lower LPD values. This result is perhaps explained by the fact that a constant volume (and thus approximate number of designs) was denied from all of these spaces and therefore is more visible in the singular dimensions to which it was applied in Figs. 22 and 24. This trend appears to continue for the other constrained design spaces although the effect is slight. In general, if a given volume is constrained within a hyperspace, the fewer dimensions which are affected, the lower the resolution (LPD) required to classify the space as Hypercubic or NHC using MI.

The next set of spaces to be examined in the experiment were also Hypercubic.

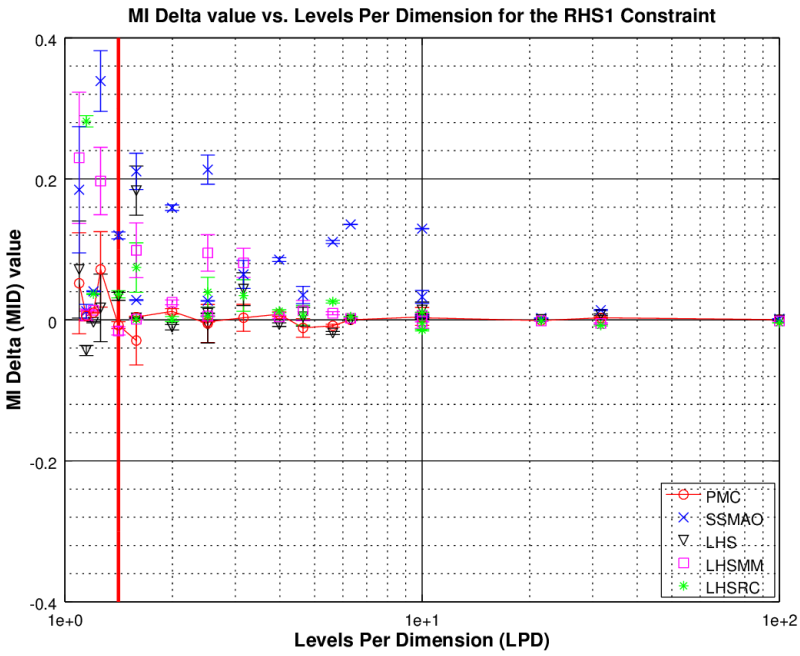

Figure 22: MI Classification Results for Reduced Hypercubic Singular 1-D (RHS1) Design Spaces

The constraints applied to these spaces did not reduce the hypervolume spanned by the design space but rather randomly removed designs throughout the space in order to simulate random failures. The application of these constraints did not introduce NHC features within the spaces. The RRFP constrained space was subjected to a random removal of a fixed 10 percent of the designs within each initial DOE, while the RRND constraint sought to remove as many cases necessary to maintain some n/d value (either 20, 30 or 50) between trials. Perhaps most noticeable is that the RRFP constrained design spaces (Fig. 25) converge to a Hypercubic Classification at much lower resolution than the RRND spaces (Fig. 26). This is because while the RRFP design spaces lose 10 percent of their cases, the RRND spaces must remove a much larger fraction of cases (in some cases over 90 percent) to maintain a given n/d ratio. Similarly to the design spaces subjected to the reduced hypercube constraints, the more structured DOEs appeared to have larger perturbations in MI values due to the

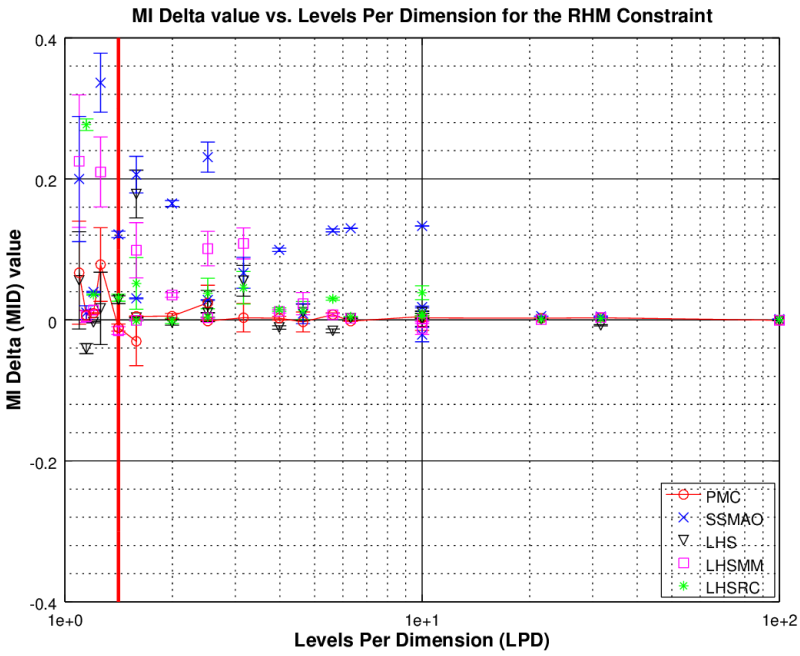

Figure 23: MI Classification Results for Reduced Hypercubic Multiple (RHM) Design Spaces

removal of designs throughout the spaces. While the PMC DOEs (pseudo-random with no structure) converge to the zero MID value expected for these Hypercubic spaces at a critical LPD value less than 2, structured DOE types such as SSMAO and LHSMM require much more resolution to ultimately correctly classify the design spaces as Hypercubic.

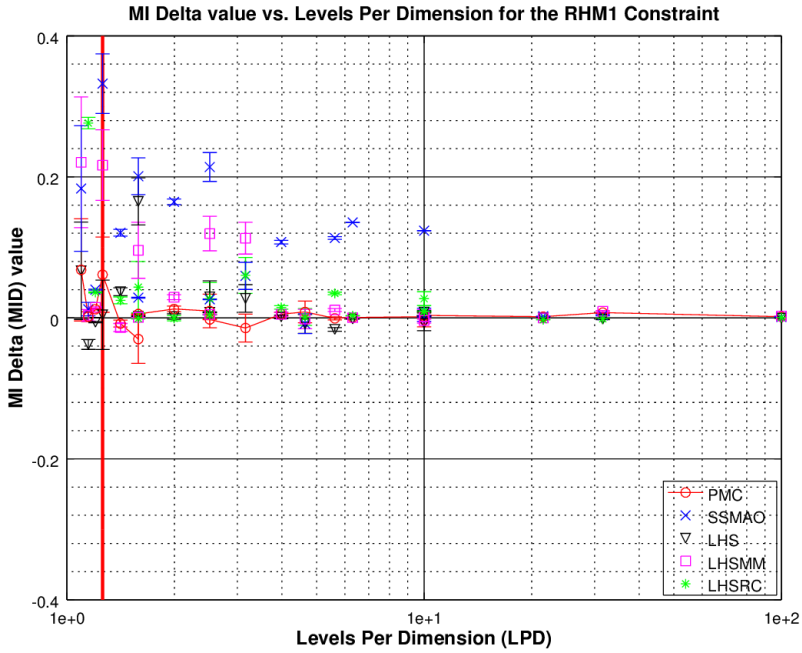

Figure 24: MI Classification Results for Reduced Hypercubic Multiple 1-D (RHM) Design Spaces

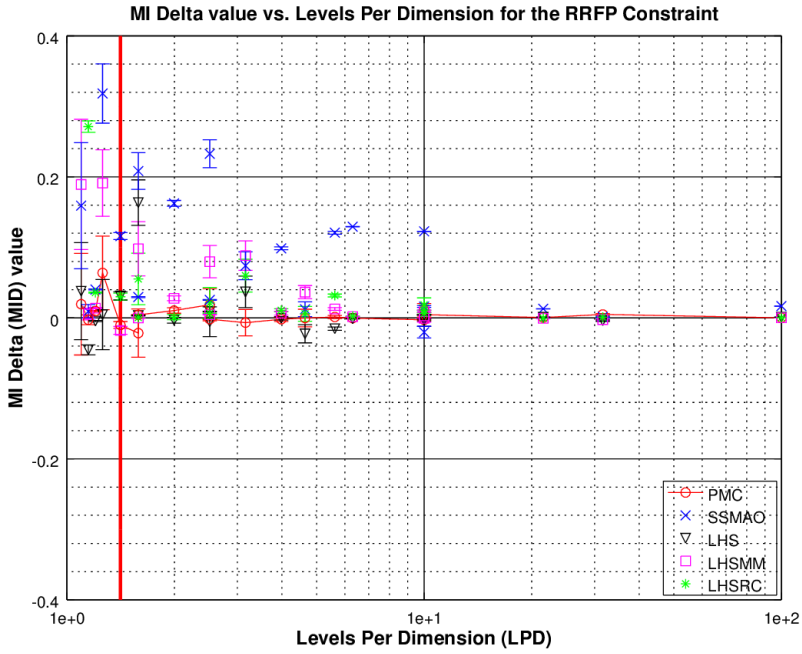

Figure 25: MI Classification Results for Hypercubic Design Spaces Subject to Random Removal of a Fixed Percentage (RRFP) of Designs

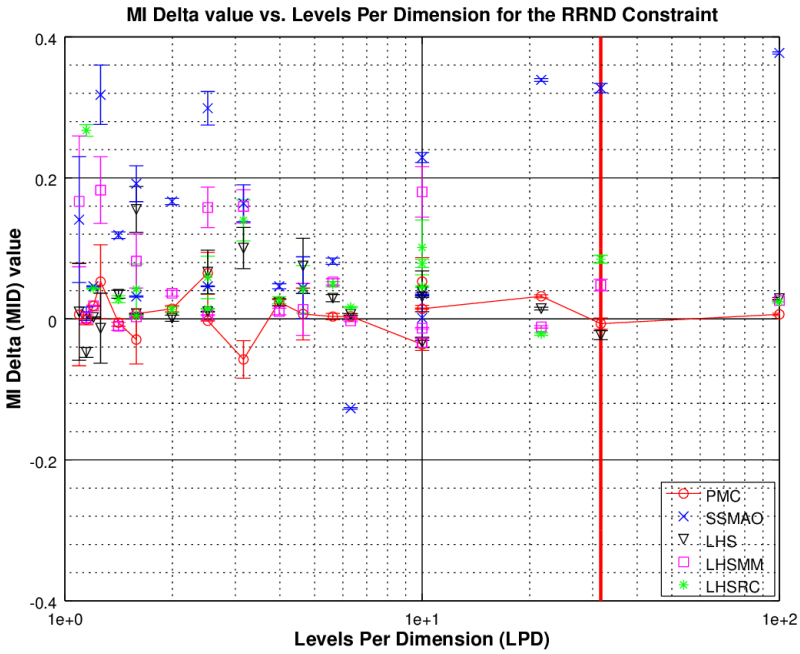

Figure 26: MI Classification Results for Hypercubic Design Spaces Subject to Random Removal of Designs to Maintain an n/d Ratio (RRND)

Figures 27 through 30 represent the results of MI tested as a Hypercubic Classifier on the first set of Non-Hypercubic constrained design spaces. These constrained design spaces featured linear constraints which eliminated 5 or 20 percent of the cases from the baseline design space in two (LCS2 and LCL2) and all dimensions (LCS and LCL). The first important trend observable in this set of results is that the MID data ultimately converge to a positive value for all of the design spaces once sufficient LPD is reached. This result serves as substantiation that MI can properly classify these spaces as NHC given sufficient resolution. Examining the results further it is clear that the constrained design spaces subjected to the large linear constraints (LCL and LCL2) ultimately converge to a higher MID values than the spaces subjected to the small linear constraints (LCS and LCS2). This seems to suggest that the spaces subjected to the large constraints, and thus featuring a larger percentage of the space denied by the constraints, are 'more' NHC than the LCS and LCS2 design spaces. Because of this, the data also appear to show that a 'more' NHC design space requires less resolution to properly classify as NHC. This is evident in the lower critical LPD values after which both the LCL and LCL2 design spaces have MID values consistently greater than zero. These results support the logical conclusion that design space features which affect larger volumes of the design space will be easier to detect using MI as a Hypercubic Classifier.

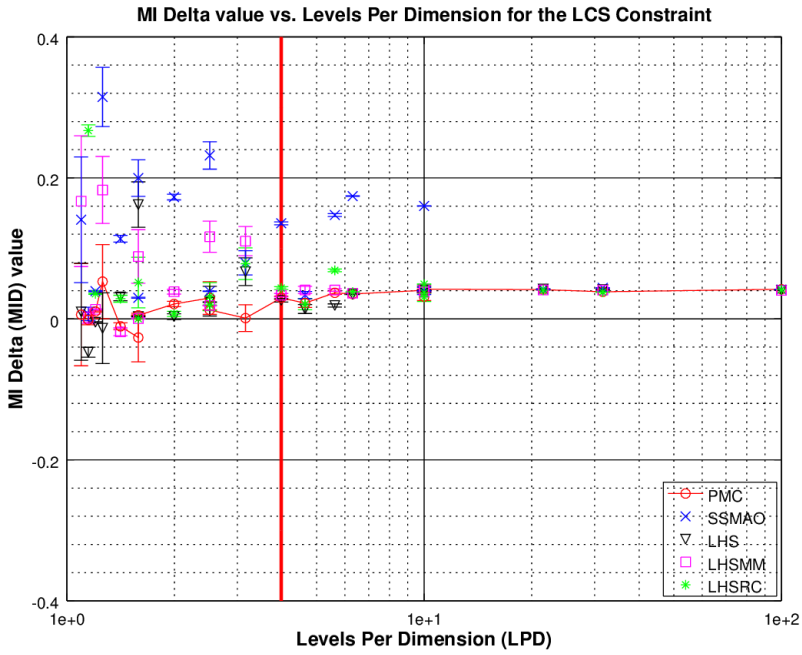

Figure 27: MI Classification Results for Non-Hypercubic Design Spaces Subject to Linear Constraints (LCS) Denying a Small Volume of the Feasible Space

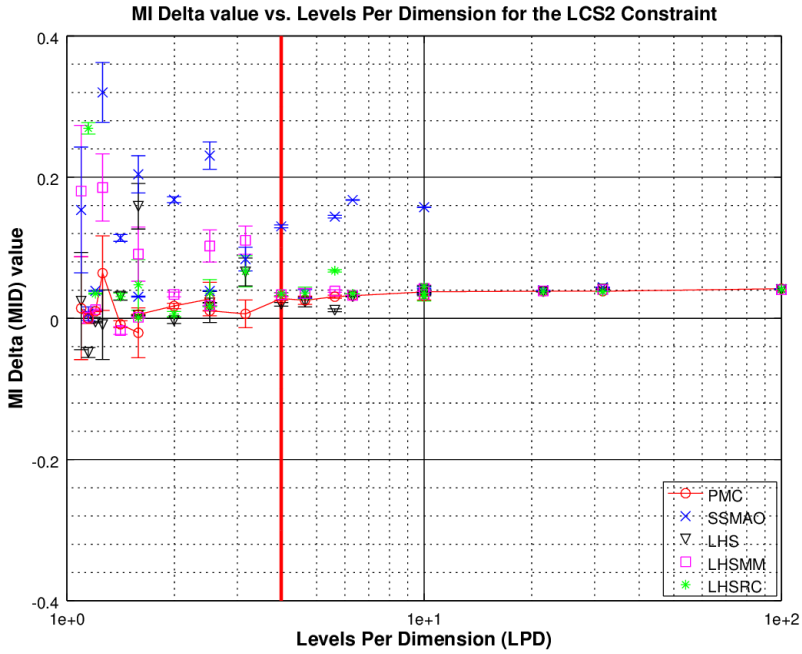

Figure 28: MI Classification Results for Non-Hypercubic Design Spaces Subject to Linear Constraints 2-D (LCS2) Denying a Small Volume of the Feasible Space

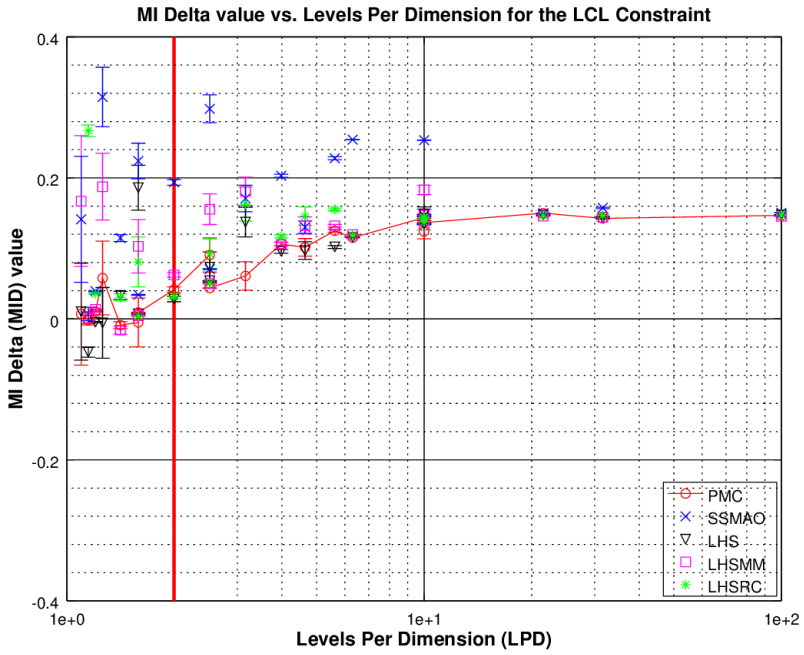

Figure 29: MI Classification Results for Non-Hypercubic Design Spaces Subject to Linear Constraints (LCL) Denying a Large Volume of the Feasible Space

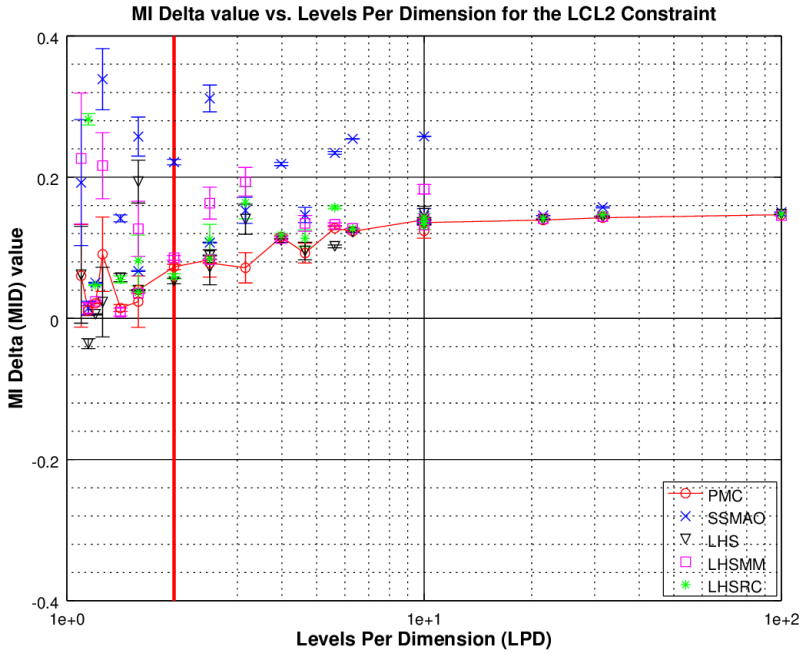

Figure 30: MI Classification Results for Non-Hypercubic Design Spaces Subject to Linear Constraints 2-D (LCL2) Denying a Large Volume of the Feasible Space

Similar trends again appear in the MI classification results when examining the non-linearly constrained design spaces (Figs. 31-34). In support of hypothesis 1, the data do indicate that MI provides proper classification of these design spaces as NHC once the resolution necessary to resolve the NHC features present is achieved within the hypervolume. Again, the design spaces subjected to the constraints which deny a larger fraction of the total hypervolume appear to be 'more' NHC which is reflected in higher MID values and a lower resolution required to definitively classify the spaces as NHC. In comparison with the linearly constrained design spaces the non-linearly constrained spaces interestingly appear to converge to similar MID values at large values of LPD. This result suggests that the ultimate MID value attained as the density of points within the original hypervolume approaches infinity may be heavily correlated with the volume of the hypervolume denied by the constraint or set of constraints which make the space NHC. This conclusion seems to make sense as in the limit if a set of variables were perfectly correlated, a very high MI value would result and the feasible design space would collapse to a line in d-dimensions of zero bounded volume.

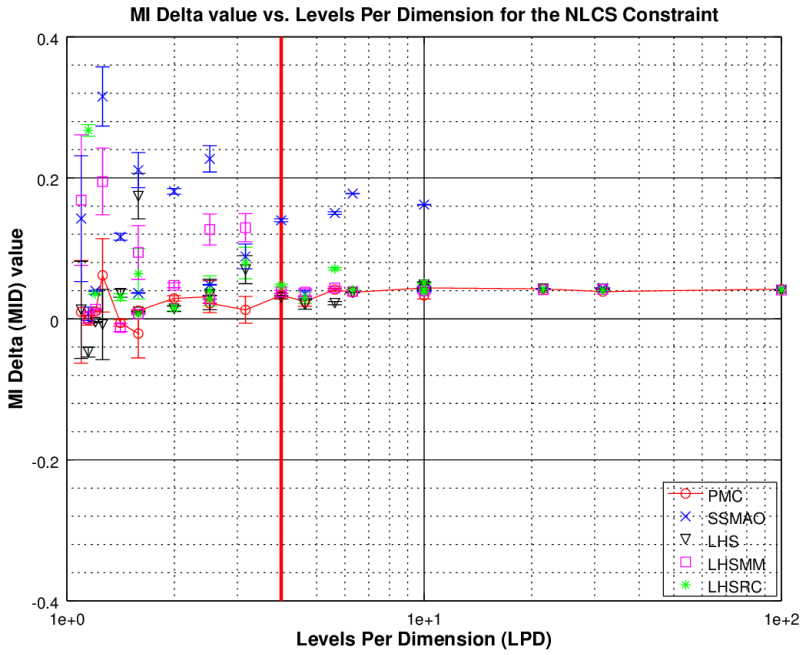

Figure 31: MI Classification Results for Non-Hypercubic Design Spaces Subject to Non-Linear Constraints (NLCS) Denying a Small Volume of the Feasible Space

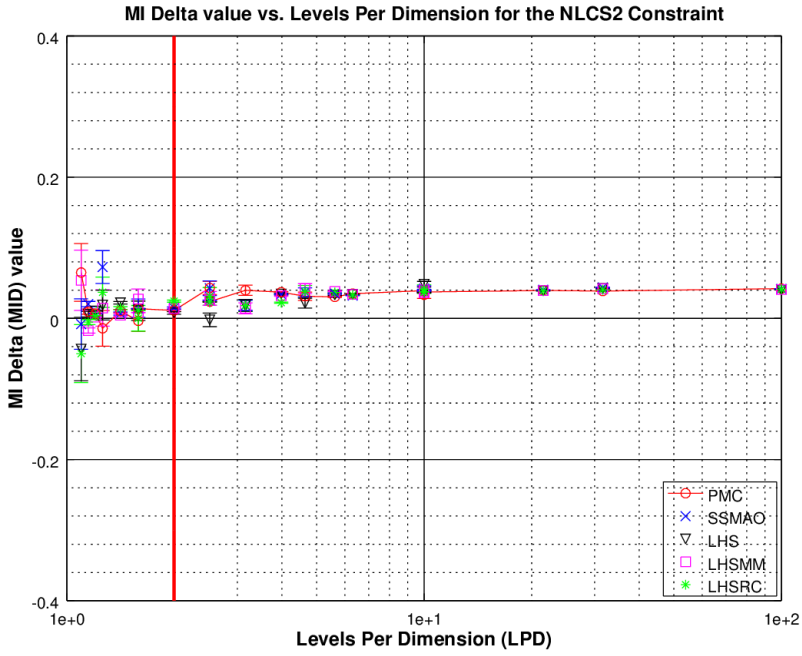

Figure 32: MI Classification Results for Non-Hypercubic Design Spaces Subject to Non-Linear Constraints 2-D (NLCS2) Denying a Small Volume of the Feasible Space

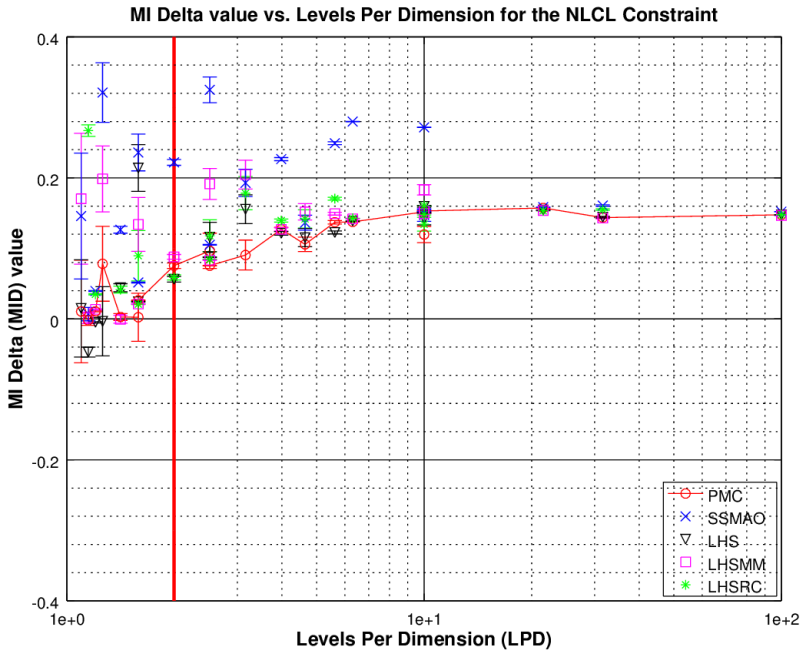

Figure 33: MI Classification Results for Non-Hypercubic Design Spaces Subject to Non-Linear Constraints (NLCL) Denying a Large Volume of the Feasible Space

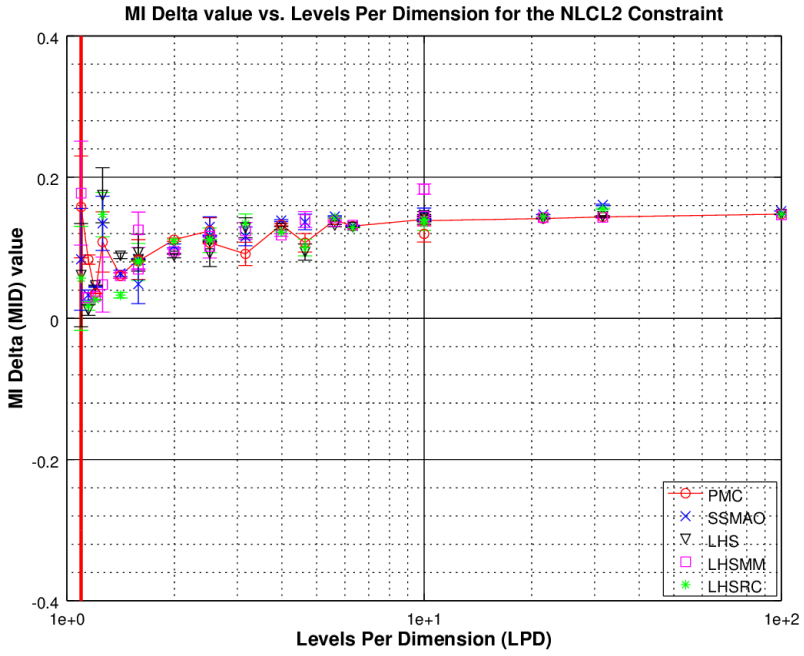

Figure 34: MI Classification Results for Non-Hypercubic Design Spaces Subject to Non-Linear Constraints 2-D (NLCL) Denying a Large Volume of the Feasible Space

The next set of constrained design spaces (Figs. 35-36) examined with MI as a Hypercubic Classifier featured hypersphere removal constraints which produced voids the center of the design spaces. These voids represented approximately 20 percent of the hypervolume bounded by the baseline Hypercubic space. This infeasible volume created by the constraints was the same percentage used for both the large linear and large non-linear constrained design spaces and as such appears to follow the similar trend of ultimately converging to an MID value of approximately 0.15 at high LPD values. A difference is again seen between the constraint applied in only two dimensions as opposed to all dimensions as it appears that the 2-D case (Fig. 36) produces a more consistent trend beginning with lower values of LPD (i.e. the data are not as dispersed about the MID = 0 line at low LPD values). This is perhaps because it is easier for MI to begin to solicit the relationship between this small subset of the design variables with low LPD as opposed to determining the correct relationship between all variables at similar LPD values. Interestingly enough however, as LPD is increased the MID value appears to increase more quickly for the spaces in which the hypersphere removal constraint is present in all dimensions.

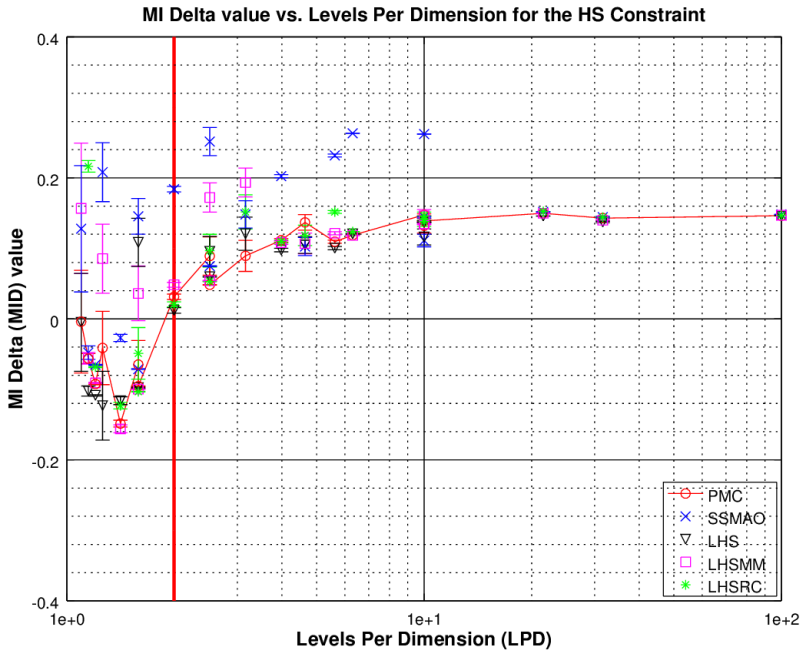

Figure 35: MI Classification Results for Non-Hypercubic Design Spaces Subject to Hypersphere Removal (HS)

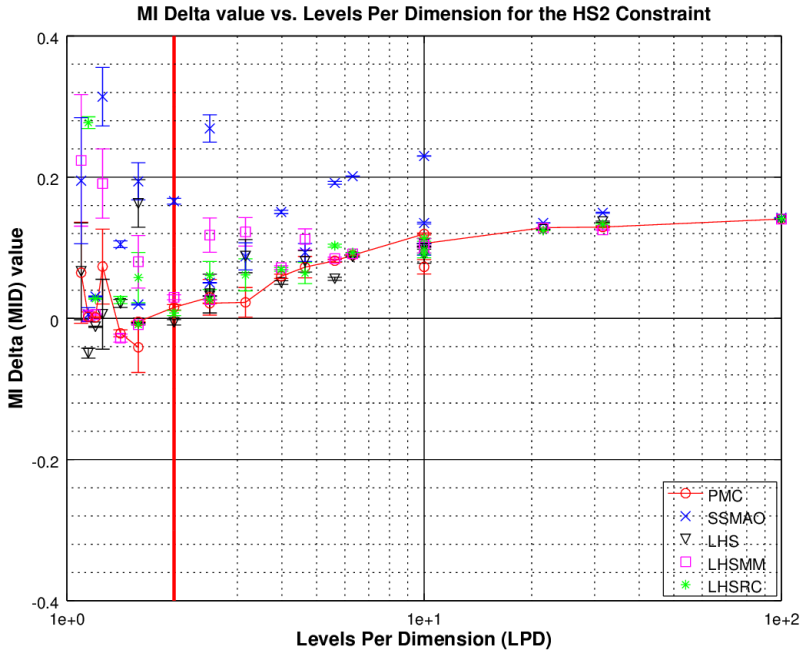

Figure 36: MI Classification Results for Non-Hypercubic Design Spaces Subject to Hypersphere Removal in 2-D (HS2)

The final set of constrained design spaces examined to evaluate the feasibility of MI as a Hypercubic Classifier were highly degenerate (likely unnatural) design spaces which produced regions of infeasibility within the design spaces according to a checkerboard pattern. As such, these spaces constructed significant correlation between design variables and thus formed NHC spaces. In all of these constrained design spaces, half of the total baseline hypervolume was made infeasible by the constraints. In the Coarse Checkerboard (CBC and CBC2, Figs. 37-38 respectively) constrained design spaces only two bins or checkers per dimension were used while for the Fine Checkerboard (CBF and CBF2, Figs. 37-38 respectively) constrained spaces featured ten bins per dimension. It is important to note that although the feasible regions demarcated by the individual bins of these design spaces are indeed Hypercubic, these spaces differ from the reduced hypercube spaces in that the entirety of their feasible regions cannot be bound (without excluding feasible regions) by simply modifying the ranges of the design variables. Perhaps what is most apparent when examining these results is that these constrained design spaces attained the highest MID values when the data ultimately converged at the higher LPD values. This follows the trend observed in the previous NHC constrained spaces which shows an increase in MID value with an increase in the volume denied within the design space by the constraints. The next but more important conclusion to be drawn from this set of results can be observed through the difference between the Coarse and Fine Checkerboard constrained design spaces. The Coarse Checkerboard constrained design spaces required very little resolution for MI to determine that these spaces are indeed NHC. This makes sense as 50 percent of the design space is denied by the constraints which also happen to produce a large correlation between design variables (this is a hard feature to miss). However, in the case of the Fine Checkerboard constrained design spaces, the MI classifier, by means of MID values remaining near zero for a significant range of LPD values, is not able to resolve the NHC nature of the design space until an

LPD of 10 or higher is attained within the space. This result is perhaps the clearest evidence of the importance of resolution for using MI as a Hypercubic Classifier. Because the CBF and CBF2 spaces had finer NHC features (10 bins in each dimension) it required many more cases spread throughout the design space to resolve the structure of these particular constraints and reject the null hypothesis that the infeasible cases dispersed throughout are simply random in nature. It is also important to note that while an LPD of 10 is quite significant (10,000 cases in only 4 dimensions) for a mid-large number of design variables and thus MI seems of limited use as a classifier in this case, the CBF and CBF2 spaces represent highly degenerate cases unlikely to be encountered in constrained design spaces for physics-based applications.

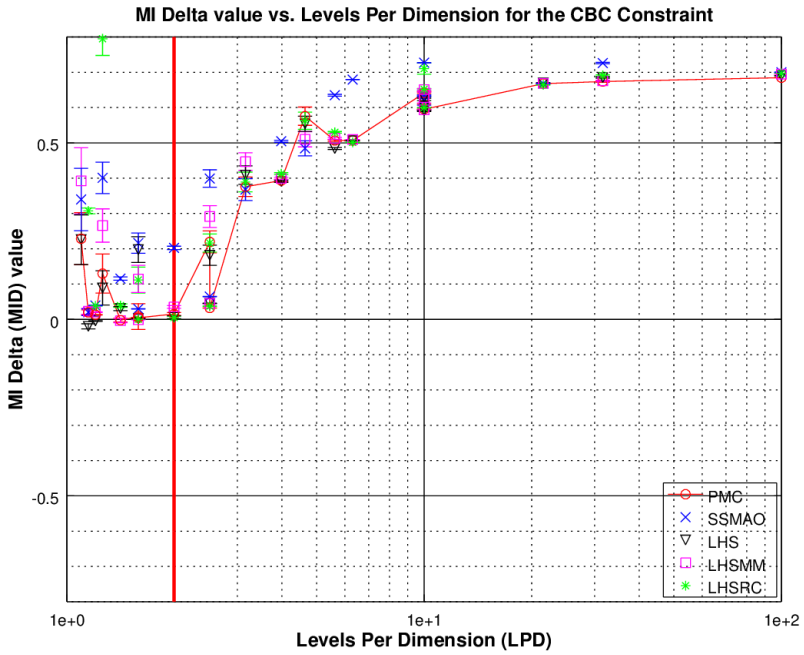

Figure 37: MI Classification Results for Non-Hypercubic Design Spaces Subject to a Coarse Checkerboard Constraint (CBC)

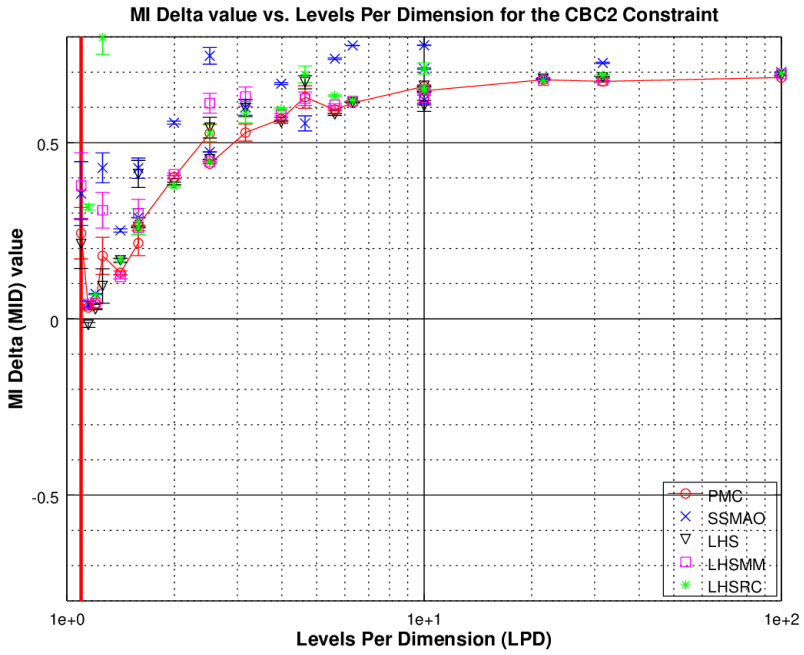

Figure 38: MI Classification Results for Non-Hypercubic Design Spaces Subject to a Coarse Checkerboard Constraint 2-D (CBC2)

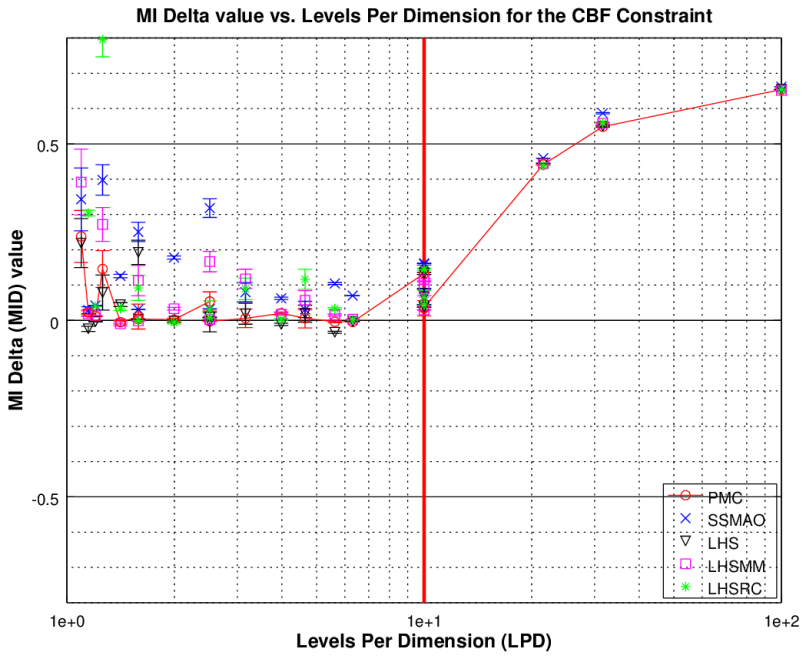

Figure 39: MI Classification Results for Non-Hypercubic Design Spaces Subject to a Fine Checkerboard Constraint (CBF)

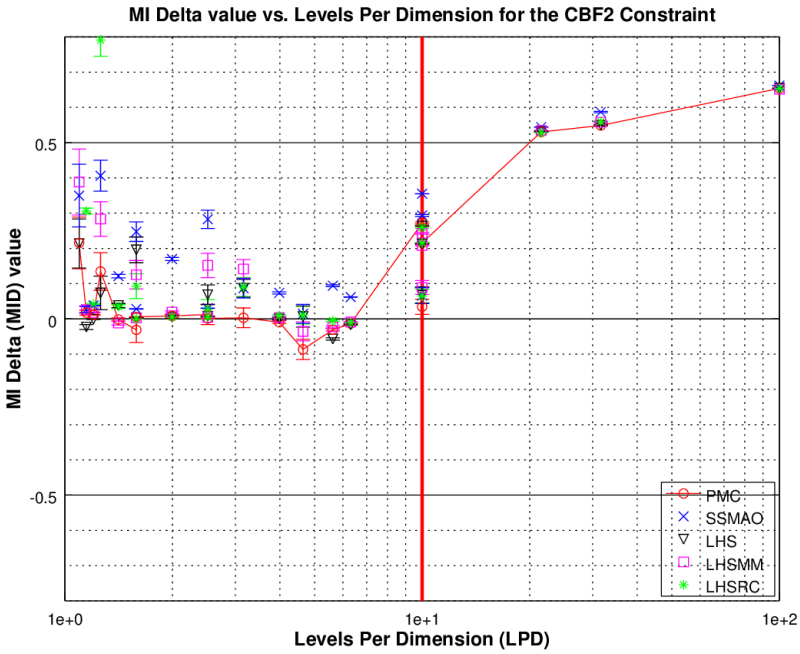

Figure 40: MI Classification Results for Non-Hypercubic Design Spaces Subject to a Fine Checkerboard Constraint 2-D (CBF2)

5.1.5 Conclusions and Consequences

Experiment 1 was performed to evaluate the utility of Mutual Information (MI) as a Hypercubic Classifier. MI was computed for unique design spaces defined by a large test matrix consisting of multiple DOE types, number of dimensions and number of cases per design space. To introduce features within the spaces which would result in either Hypercubic or NHC design spaces, 20 separate constraints were applied to each unique baseline design space created from the test matrix. To test the hypothesis that MI could serve as a Hypercubic Classifier, MI values were computed for each unique baseline unconstrained design space and then again for the 20 constrained variants of that space. This process was repeated for 60 replicates for each unique space. These MI results were combined through a metric termed 'Mutual Information Delta' or 'MID' (Eqn. 27).

Under the original procedure of the experiment, a space was to be classified as Non-Hypercubic if the mean MID value minus the lower bound of the standard error about the mean over all the replicates was greater than zero. Conversely a space would be deemed Hypercubic should the MID mean including standard error not depart from zero. Through the course of the experiment it was found that MI could provide such classifications and do so accurately for both the Hypercubic and NHC, but only with the caveat that sufficient resolution existed within the hypervolume to resolve the features imposed by the constraints.

Because of this resolution requirement for MI to provide accurate Hypercubic Classification, a similarity parameter was sought to collapse the data observed in all of the design spaces examined of differing number of dimensions and number of cases. First tested was n/d, a common similarity parameter utilized to ensure computer experiments have sufficient resolution to produce accurate regressions. However as the hypervolume defined by the design spaces increased exponentially with the addition

of dimensions this similarity parameter proved ill suited toward establishing a criterion for the resolution required to resolve NHC features within d-dimensional design spaces. To address the gap a new similarity parameter coined 'Levels Per Dimension' or 'LPD' (Eqn. 30) was devised.

This new similarity parameter was compared against n/d and found to be much more appropriate for determining threshold resolution values beyond which MI could be used as a reliable Hypercubic Classifier. Using LPD as a measure of metric it was discovered that the larger a NHC feature was, i.e. the more volume that was denied by the constraint in the original Hypercubic space, the less resolution required to resolve the space as NHC. Interestingly enough, the type or shape of the constraint (linear, nonlinear, void, etc.) was not as significant for classification purposes as the volume denied, or corresponding number of cases made infeasible, by the constraint.

Also revealed by the experiment was the effect that different types of space filling DOE approaches had on the ultimate MI classification results. Although all DOE types tested were space-filling in nature, certain designs were perhaps more appropriate for use with the MI classifier than others. Observed especially with the SSMAO and LHSRC based design spaces, these more structured designs could lead to incorrect conclusions as cases were removed unless these conclusions were drawn at really high resolution (LPD). This is likely because even random removal of cases within these design spaces greatly affected the distribution of cases within the space and thus manifested as large MID values. As such, these types of design spaces tend to provide false positives by classifying Hypercubic spaces as NHC, unless LPD is very high. This effect is less pronounced in DOE types which are closer to 'true' random and independent dispersal of cases such as PMC. This result is important as it shows that these space-filling design types are all compatible with using MI as a Hypercubic Classifier, yet their selection greatly affects the resolution required for MI to return a correct classification of the design space. The following table (Table 5) indicates

the approximate critical resolution expressed in LPD value (with a tolerance of ±0.01 in MID with error about zero) after which each constrained design space could be correctly classified using MI.

Table 5: Critical LPD Values Required to Correctly Classify Design Spaces Using Mutual Information

Design Space Characteristics		Critical LPD Values by DOE Type				
Constraint ID	SCP	PMC	SSMAO	LHS	LHSMM	LHSRC
RHS	0.80	2.512	100.000	4.642	4.642	6.310
RHS1	0.80	1.413	100.000	3.981	3.981	6.310
RHM	0.80	1.413	21.544	1.995	3.981	6.310
RHM1	0.80	1.259	21.544	1.995	3.981	6.310
RRFP	0.90	1.413	31.623	1.995	6.31	6.310
RRND	0.01 - 0.60	31.623	N/A	N/A	N/A	N/A
LCS	0.95	3.981	1.202	3.162	1.995	2.512
LCS2	0.95	3.981	1.202	3.162	1.995	2.512
LCL	0.80	1.995	1.202	1.995	1.995	1.995
LCL2	0.80	1.995	1.202	1.413	1.585	1.096
NLCS	0.95	3.981	1.202	1.995	1.995	1.995
NLCS2	0.95	1.995	1.995	3.162	3.981	3.981
NLCL	0.80	1.995	1.202	1.413	1.585	1.096
NLCL2	0.80	1.096	1.096	1.202	1.413	1.148
HS	0.80	1.995	1.995	1.995	1.995	1.995
HS2	0.80	1.995	1.202	2.512	1.995	1.995
CBC	0.50	1.995	1.202	2.512	1.995	2.512
CBC2	0.50	1.096	1.096	1.202	1.096	1.096
CBF	0.50	10.000	5.623	10.000	10.000	10.000
CBF2	0.50	10.000	5.623	10.000	10.000	10.000

A few conclusions can be drawn from this set of tabulated data. Firstly, regarding

DOE selection, structured designs can be problematic for use in MI Hypercubic Classification. While more structured DOE types such as SSMAO, LHSMM and LHSRC may require in general lower critical LPD values to correctly classify the NHC spaces, they require a great deal of resolution in order to correctly classify the Hypercubic spaces. This means should they be used for classification they may lead the user to believe almost any design space with a SCP less than 100 percent to be NHC. Therefore, while the PMC DOE type has higher critical LPD values in order to resolve NHC spaces compared to these designs, it is much less likely to generate false positive results when used to classify Hypercubic spaces. Secondly, with the PMC DOE type it appears that features in a reduced number of dimensions can be resolved at lower LPD values than their all-dimensions counterparts. This trend is not consistent across DOE types however. Lastly, it appears that non-linear constraints or features were more difficult to resolve than linear constraints which yielded the same SCP, yet SCP itself was a major driver in critical LPD. This reflects the earlier observation that the greater the volume of the space denied by a NHC feature, the 'more' NHC the space is and thus the easier it is for MI to provide correct classification at lower resolution.

Ultimately the results of Experiment 1 substantiate Hypothesis 1 and illustrate that Mutual Information (MI) is an appropriate Hypercubic Classifier. This statement is bound however by the requirement that the space be sampled with sufficient resolution, quantified in this experiment through equivalent Levels Per Dimension (LPD), to resolve whatever features may exist within the space. The question of what LPD is needed is not a trivial one and is dictated by the type of DOE used to sample the space as well as the shape and volume of features that exist within the space. Table 5 provides guidance in selecting this required LPD, however in general, the largest resolution which is affordable should be used should LPD available be less than 2.

5.2 Experiment II

5.2.1 Motivation and Thought Experiment

As a means to classify a given design space as Hypercubic or NHC was established through the results of Experiment I, the next logical question was how can such information be leveraged to improve the understanding and future exploration of this space? The focus of this experiment is therefore to ascertain whether having a bounding of the design space, constructed from knowledge obtained with an initial design space exploration, is beneficial when revisiting the design problem. While this experiment may seem able to be proven by logic alone, depending on the shape of the Non-Hypercubic design space and the quality of the bounding generated, such a conclusion may not be so trivial. Furthermore, should the design space lack sufficient LPD in its initial sample, a bounding constructed from such information may incorrectly infer non-existent characteristics within the space.

In order to test hypothesis 2, multiple Non-Hypercubic design spaces of different shapes, characteristics and dimensions were initially sampled and bound. They were then re-explored with and without the use of their respective boundings. Two competing methods for repeat design space exploration were evaluated within this experiment. The first method meant to provide a baseline respective of current practices is termed All At Once AAO sampling in which no changes are made to the structure of the DOE between the initial sample and the final sample. This approach is equivalent to a DOE in which all samples were taken initially. The second method, termed Bounded Adaptive Sampling BAS, leverages a bounding for targeted re-sampling of the design space after the initial sampling has been performed. The hypothesis that a bounding is generally beneficial is considered substantiated if and only if the use of a bounding in each case allows for the more efficient re-exploration of the design space.

5.2.1.1 *Hypothesis 2: Regarding RQ2*

If a bounding is constructed using sufficient resolution to resolve the features present within a Non-Hypercubic design space, then it can be leveraged to enable more resource efficient future exploration of the space.

5.2.2 Experiment Design

In order to confirm that the use of a bounding for the feasible space would aid in adaptive sampling of design spaces, multiple NHC spaces of differing dimension and sampling resolution were examined. For the hypothesis presented in this experiment to be substantiated, the BAS approach (when provided sufficient resolution) must reliably outperform the AAO sampling without being prohibitively expensive in terms of computational resources. To provide variety in the design spaces examined, PMC sampling was used to populate the baseline design spaces which were then made NHC by the application of 14 separate constraints which produced features which could appear in generic design spaces.

5.2.2.1 *Apparatus*

To determine if a BAS approach was useful in enabling further DSE of NHC spaces, some representative NHC test design spaces were again required. As only NHC constrained design spaces were needed for this experiment, only the constraints which would yield NHC feasible design spaces were utilized. Again, these specific constraints were applied numerically (See Appendix for details) to the baseline Hypercubic design spaces defined for each repetition of each unique design space sample (combination of settings from the test matrix). These constraints would be queried again once adaptive samples had been generated with the use of the bounding classifiers to determine which designs among the adaptive samples were feasible with respect to the relevant constraint.

As this experiment was meant to examine the utility of using a bounding of a

feasible design space for adaptive sampling purposes, a means of constructing such a bounding was necessary. The specifics of the bounding method were largely unimportant for this experiment. Any type of method used to enable bounded adaptive sampling (BAS) was sufficient to simply compare against the common approach of performing design space exploration all at once (AAO) with a DOE which disperses the entire case budget throughout the Hypercubic design space defined by the limits on the design variables. As such, the randomForest package implemented in the statistical programming language R was utilized to construct random forest (RF) classifiers for the purpose of bounding feasible design spaces for this experiment [83, 23]. RF was selected as the bounding method for this experiment due to its ease of implementation, interpretation and speed of fitting within R. A unique random forest classifier was fit for each unique constraint defined feasible set, however all of the random forests were composed of 2000 different decision trees.

5.2.2.2 Metrics

As the ORO seeks a resource efficient bounding of the feasible design space, it was necessary to track some measure of the computational expense required to construct and use such a bounding. This measure could be utilized to then to quantify the cost increase compared to sampling BAU without a bounding for the feasible space. Furthermore, a measure of the benefit provided by the use of the BAS to guide DSE was also desired to justify its adoption. Therefore, two metrics were utilized in this experiment for the purposes of assessing the efficiency of the two competing methods for repeat design space exploration. They are as follows:

- **Method Execution Time** MET: the computational time (measured in seconds) required for the respective methods to generate their required elements. For the AAO case, this will only include the time required to create the sampling DOE (PMC sample set). For the BAS case also included will be the time

required to create the bounding and the time required to create the bounding influenced sample set.

- **Feasible Design Ratio** FDR: the ratio of feasible designs obtained within the sample set to the total number of design evaluated within the sample set. This metric measures how successful each sampling method is at returning feasible designs.

5.2.3 Experiment Settings and Execution

This experiment required Non-Hypercubic design spaces to be evaluated and bound. However, as the shape and characteristics of the design space to be evaluated are often not known a-priori, it was important to evaluate a number of different design spaces to show the general applicability of the hypothesis. For this reason, multiple design spaces were be tested in this experiment and these tests replicated to reduce random error. Design space shapes and characteristics desired for testing were as follows: slightly Non-Hypercubic, highly Non-Hypercubic, discontinuous, convex and non-convex. In an attempt to remove the influence of sample set structure, PMC sampling was used for all design spaces and the sample size was varied for different trials. A reduced test matrix from that featured in Experiment I was used to define a unique design space for each trial, it is enumerated below:

5.2.3.1 Test Matrix

- **DOE**: Pseudo-Monte Carlo (PMC)
- **Dimensions (d)**: 2, 3, 10, 20, 50
- **Cases (n)**: 100, 1000
- **Constraints**: Hypersphere (HS), Hypersphere 2-D (HS2), CheckerBoard Coarse (CBC), CheckerBoard Coarse 2-D (CBC2), CheckerBoard Fine (CBF), Checker-Board Fine 2-D (CBF2), Linear Constraint Small (LCS), Linear Constraint

Small 2-D (LCS2), Linear Constraint Large (LCL), Linear Constraint Large 2-D (LCL2), Non-Linear Constraint Small (NLCS), Non-Linear Constraint Small 2-D (NLCS2), Non-Linear Constraint Large (NLCL), Non-Linear Constraint Large 2-D (NLCL2) (See Appendix)

5.2.3.2 Procedure

While ensuring sufficient replications were included to have confidence in the results, it was desired to minimize the number of replications for each trial, as each additional replication required the training of an additional bounding classifier. To quote Gauch, "Replication is one of the finest ideas in science, but it faces a severe law of diminishing returns" [27]. Choosing 10 replications per trial, allowed for the observed result to be more accurate than a single observation 80.5 percent of the time. This was deemed sufficient to capture the general behavior of the bounding classifiers and observe, with reasonable confidence, if a BAS approach was superior to sampling AAO. For each replication, a bounding was constructed using information from the initial design space exploration as training data. The space was then be re-explored for each replication following both AAO and BAS practices. Metric values were then computed for both the AAO and BAS approaches and compared. The mean value of the metrics over all 10 replications allowed conclusions to be drawn about the superior method for design space re-exploration for those trial settings. The enumerated procedure is as follows:

1. Select from the test matrix a unique combination of number of dimensions and number of cases - this defines a design space
2. Generate 10 replications of this design space using PMC sampling
3. For each replication, individually apply each constraint and save the corresponding NHC feasible design spaces which result in a series of constraint defined feasible sets (CDFS)

4. compute the FDR for these constrained initial samples (what is the ratio of feasible cases to total cases for each constraint defined feasible set)

5. Using the initial sample as a training set, fit a random forest classifier to construct a bounding for each constraint defined feasible set and then use this classifier to suggest another n cases it predicts will be feasible within this bounding

6. For the baseline sample, simply randomly select another n points (effectively ignoring the presence of the constraint, this simulates the business as usual practice of a random resampling or simply expending the entire budget All At Once AAO)

7. Re-explore the design space using these new sample sets performing both BAS and AAO and evaluate the FDR for each

8. Average and record the MET and FDR across all of the 10 replications for both the AAO and BAS enabled re-sampling of the design space

9. Repeat steps 1-8 for each unique design space (unique combinations form the test matrix)

10. Draw final conclusions

In order to draw final conclusions in this experiment for each unique design space, the successful percentage of cases in the new sample resulting from AAO vs. BAS were compared. The hypothesis was considered substantiated if BAS consistently provided a higher successful percentage (FDR) than AAO in the new samples.

5.2.4 Results Discussion

The first result to emerge from experiment 2 concerning the MET was important if underwhelming. Table 6 summarizes the MET breakdown for the two competing methods of sampling the feasible design spaces. Although quantifiable, the difference

in MET between AAO and BAS within this experiment is considered negligible. This is nevertheless a significant result as it demonstrates that the computational resource expenditure required to use the methodology (and construct boundings for high-dimensionsal design spaces) may not be prohibitive. The boundings constructed in this experiment are simplified versions of what would likely be used in practical applications, however this result shows that at least in these cases, the performance of BAS itself was not a showstopper. To provide a hardware benchmark, these tests were performed on a system running 64-bit Windows 10 with an Intel Core i7-4790k processor at 4.00 GHz with 16 GB of RAM.

Table 6: Summary of Method Execution Time (MET) Required for AAO and BAS

	Training of Bounding Classifier		Generation of New Points	
Sampling Method	min time (s)	max time (s)	min time (s)	max time (s)
AAO	N/A	N/A	0.000	0.006
BAS	0.067	11.270	0.000	1.306

Results Interpretation (Figs. 41-54) The reduced test matrix utilized in experiment 2 produced 10 distinct baseline design spaces ranging in initial sample resolution from LPD values of 1.10 to 31.64. Figures 41-54 illustrate the results of the AAO vs. BAS tests performed on these spaces measured in the difference in FDR between AAO and BAS. Two horizontal reference lines are included in each of the figures to illuminate the possible range of FDR improvement. The upper reference line is defined by the amount of baseline design space volume denied by the given constraint the space is subject to while the lower line represents a net zero improvement over AAO. If the FDR delta was positive (above the zero reference line in the figures) for a given design space then there was a greater number of feasible cases achieved in the second sample set through the use of BAS. This type of result would indicate that the use of a bounding to further explore the feasible design space was beneficial. If the FDR

delta approaches the upper reference line then the BAS method is suggesting almost entirely feasible cases, suggesting that the bounding fit to the feasible design space may be accurate. It is important to note however that a FDR approaching unity does not itself guarantee bounding accuracy. For example, a conservative bounding for the space could have been constructed which while eliminating most false positive suggestions for feasible designs may reject many truly feasible designs as false negatives. The variance which can be observed in the box plots illustrates variability in both the unique PMC designs over the the ten replicates for each constrained design space type and the bounding RF classifiers fit to them. This variance is generally seen to decrease with increasing resolution as the PMC replications more closely resemble each other in terms of design case distribution throughout the design space with a denser sample. Additionally, with increasing resolution, the RFs fit to each replication generally begin to better represent the true boundary and thus also begin to more closely resemble one another. However, as LPD is not a perfect measure of resolution this trend is not always monotonic. Additionally, some very low variances in results near zero FDR delta can be seen at very low resolution. This is indicative that the RFs fit for these spaces are so inaccurate that they classify nearly every point within the space as feasible and thus hardly differ from AAO sampling results.

The first NHC design spaces examined in this experiment were those whose corresponding feasible spaces were determined by linear constraints (LCS, LCS2, LCL, LCL2). Figures 41-44 depict the improvement in FDR gained through performing BAS over AAO for design spaces sampled with differing resolutions measured in LPD. It is important to note that while a valuable similarity parameter, LPD is not a perfect measure for resolution as can be seen in these plots. Two separate design spaces have an LPD equal to 10 (n = 100, d = 2 and n = 1000, d = 3) yet the results for these distinct spaces differ even though their LPD does not. In general, for design spaces with similar LPD values, the space with the higher n/d allows for features

to be resolved more easily and thus tends to have more accurate boundings for the feasible space. This result is perhaps obvious but interesting nonetheless, suggesting that resolution might be best represented by tracking both LPD and n/d.

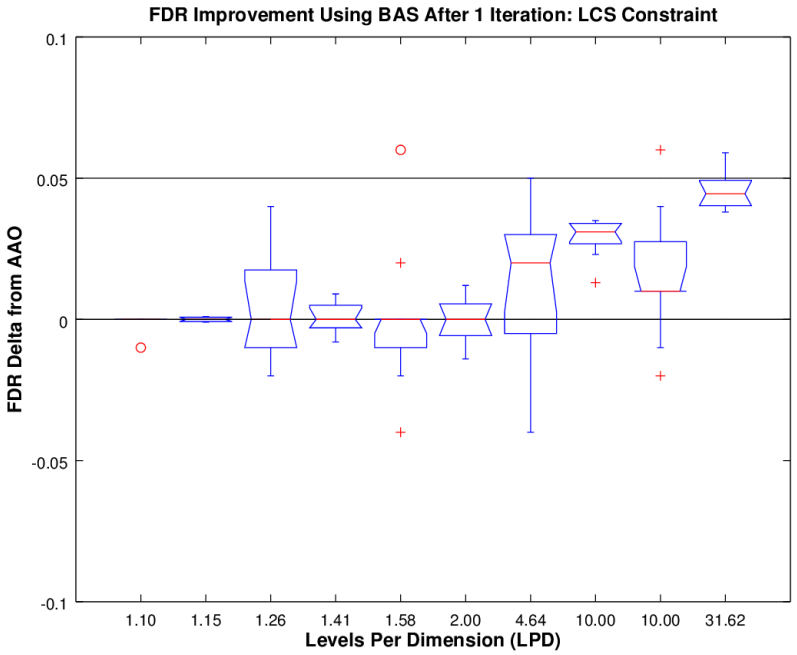

Figure 41: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Linear Constraints (LCS) Denying a Small Volume of the Feasible Space

Again, similar to the use of MI as a classifier, resolution plays an important role in determining whether or not a bounding constructed from an initial sample of a constrained design space and then used for adaptive sampling will provide an advantage over the common practice of sampling AAO. Figures 41 and 43 illustrate that an LPD of 2 or higher is needed for a bounding to be constructed with sufficient accuracy to improve the FDR using BAS over that which could be achieved with AAO sampling. For the design spaces subjected to constraints only in two dimensions however (Figs. 42 and 44), a beneficial bounding can be constructed at much lower resolution. It is also very important to observe that perhaps contrary to intuition, using a bounding may actually be worse in some situations than sampling blindly

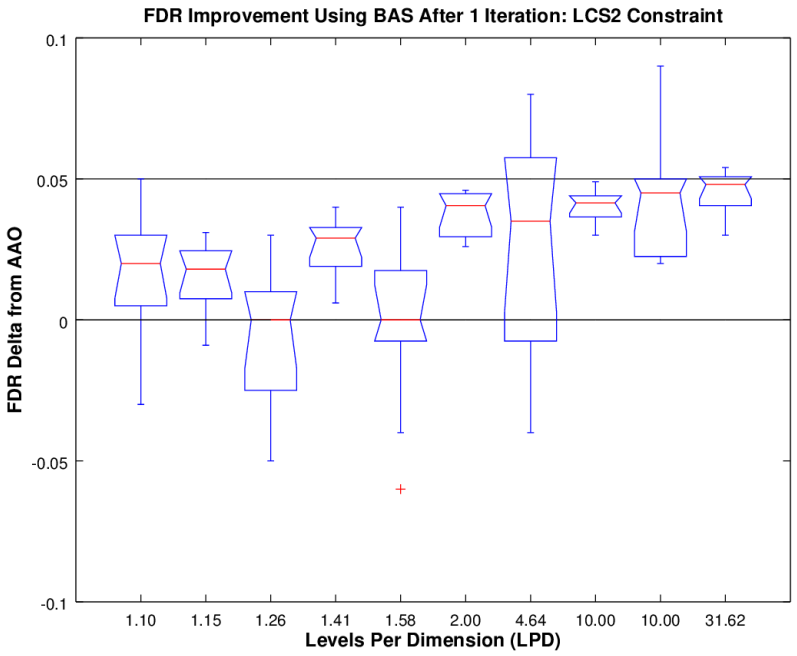

Figure 42: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Linear Constraints 2-D (LCS2) Denying a Small Volume of the Feasible Space

following AAO practices. This result only appears in cases where the design space was sampled with very low resolution and thus an inaccurate bounding is constructed for BAS, yet it clearly illustrates that using a bounding does not always provide a superior way of sampling.

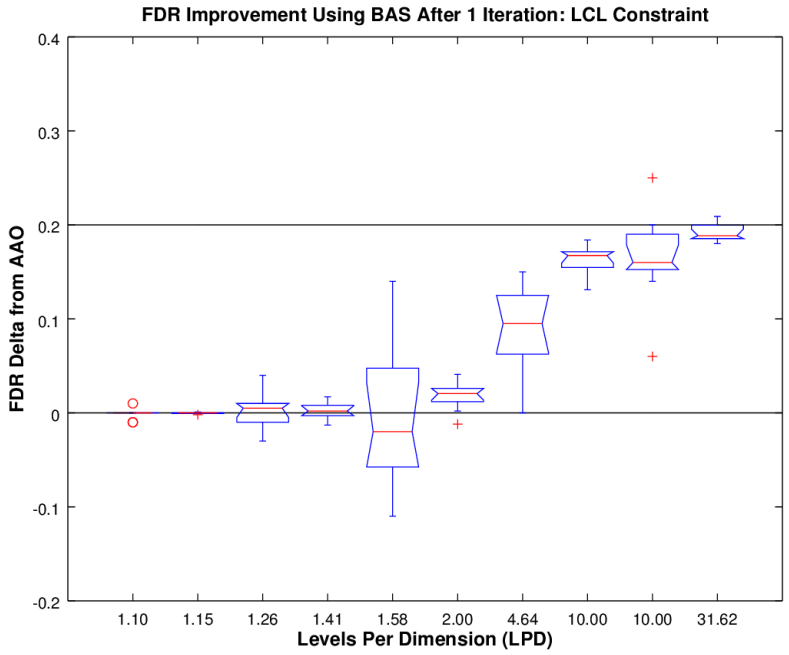

Figure 43: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Linear Constraints (LCL) Denying a Large Volume of the Feasible Space

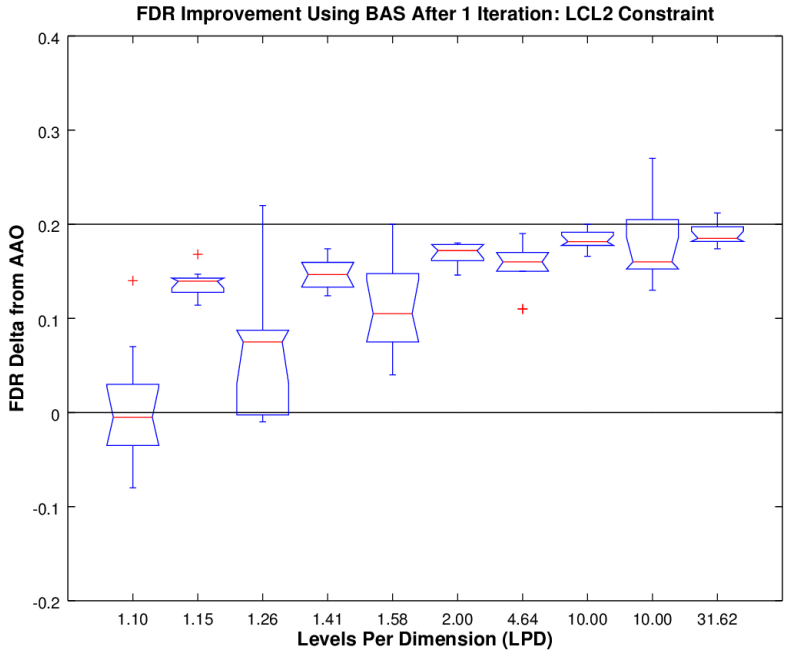

Figure 44: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Linear Constraints 2-D (LCL2) Denying a Large Volume of the Feasible Space

Similar trends emerge when examining the results for the Non-Linearly constrained design spaces. Interestingly, these spaces seem to have slightly more accurate boundings for the same LPD over their linearlu constrained counterparts. This makes sense as the decision trees composing the random forests constructed as boundings for these spaces would be able to approximate the non-linear constraint with greater accuracy with a given number of branches. Another interesting effect, seen most prevalently in the 2-D constrained design spaces (Figs. 46 and 48), is that n/d in addition to LPD seems to be driving the accuracy of the boundings and thus their effectiveness in suggesting new feasible designs. This effect is likely arising due to the fact that for a 2-D constrained design space a relevant bounding need only be fit to the two NHC dimensions. Thus a higher dimensional space can be seen as effectively collapsed from the point of view of the bounding classifier and thus the LPD is artificially increased due to the collapse of these dimensions. It is important to note that these dimensions do not disappear, they are simply irrelevant for the bounding and thus the NHC design space can be thought to collapse to a 2-D space in the two NHC variables.

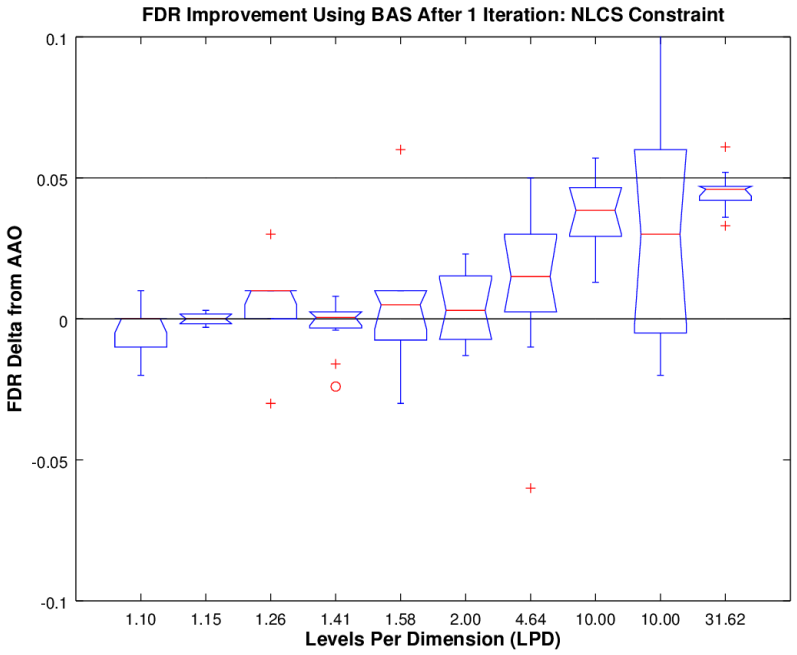

Figure 45: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Non-Linear Constraints (NLCS) Denying a Small Volume of the Feasible Space

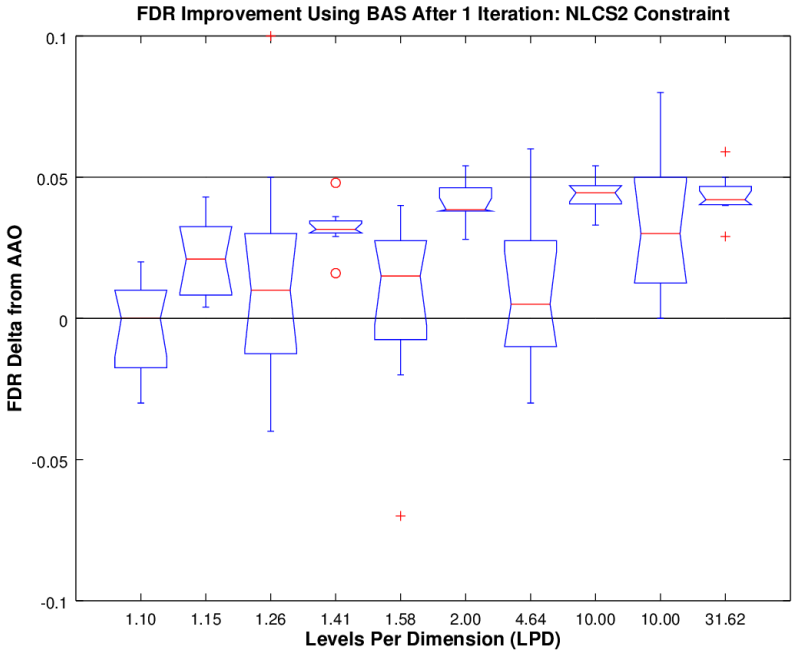

Figure 46: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Non-Linear Constraints 2-D (NLCS2) Denying a Small Volume of the Feasible Space

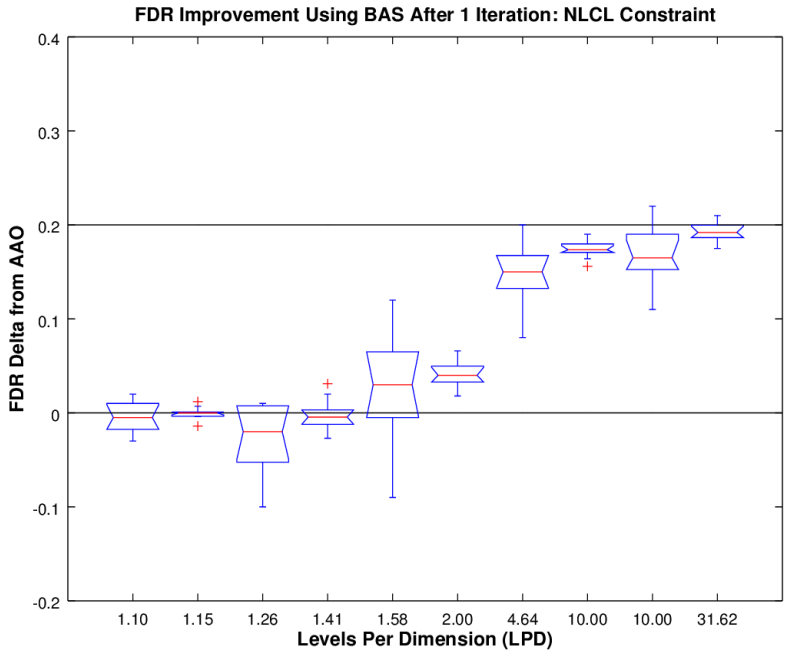

Figure 47: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Non-Linear Constraints (NLCL) Denying a Large Volume of the Feasible Space

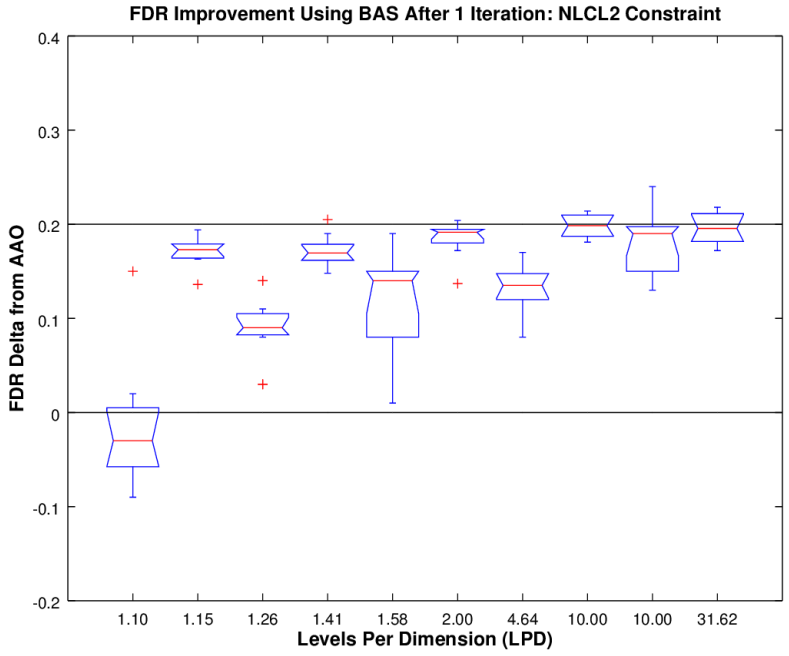

Figure 48: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Non-Linear Constraints 2-D (NLCL2) Denying a Large Volume of the Feasible Space

The FDR improvements as a result of BAS for the design spaces subject to hypersphere removal seem to lie in-between the results for the linearly and non-linearly constrained spaces. Again, this could be due to the nature of the random forest classifier used to construct boundings and the ease at which it can approximate features that exist within a space. Resolution still plays a very important role and it is still clear that a bounding can produce a net negative result compared to AAO sampling at very low resolution values.

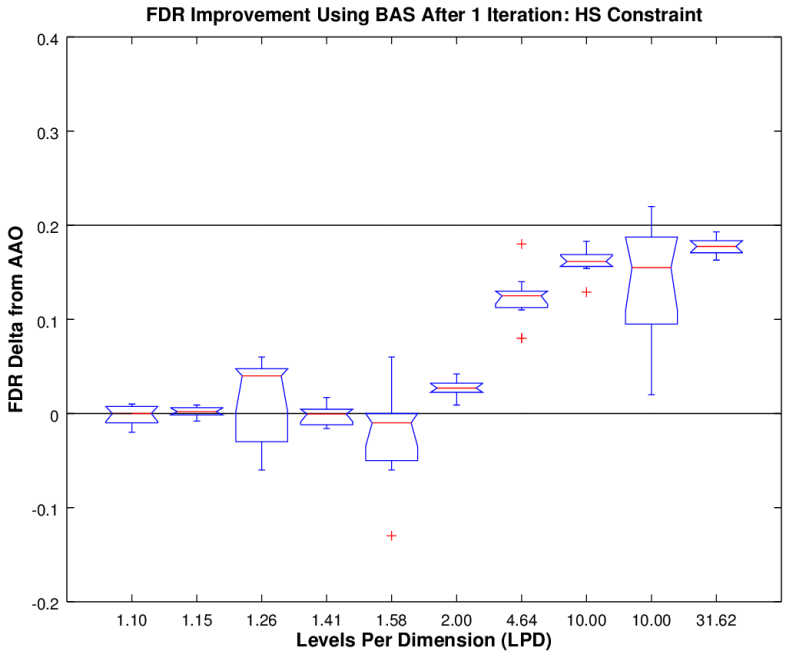

Figure 49: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Hypersphere Removal (HS)

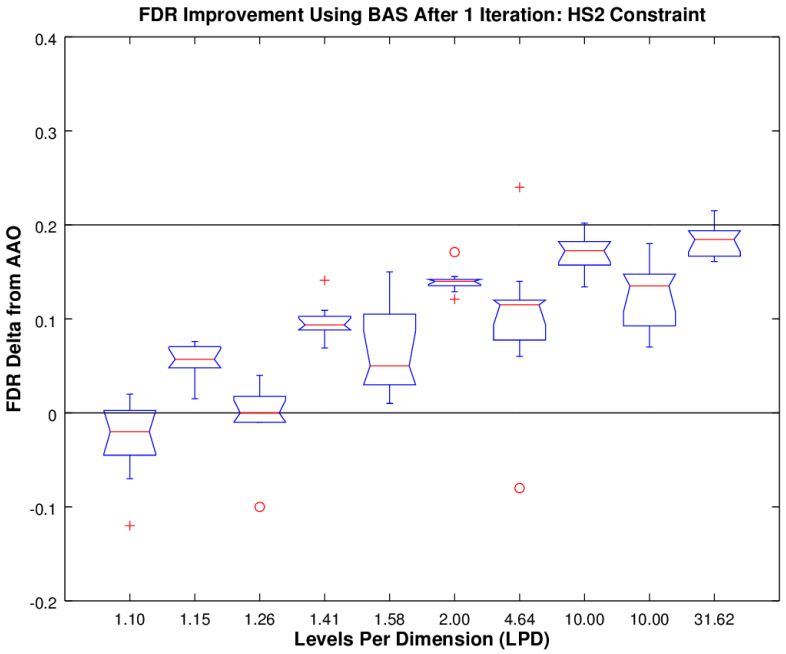

Figure 50: Effects of Bounded Adaptive Sampling for Design Spaces Subject to Hypersphere Removal in 2-D (HS2)

The trends remain evident when examining the Checkerboard Constrained design spaces, yet the jump in FDR once sufficient resolution was attained was much more pronounced. This effect is again likely due to the use of random forests as the bounding classifiers as decision trees are particularly apt at resolving the shapes formed by the checkerboard constraints. Another trend may also be observed looking back at the critical LPD values presented in Table 5. In all of the figures, a net positive result in FDR improvement over AAO sampling can be seen for resolutions above the corresponding critical LPD values for the constrained PMC design spaces. This makes sense as MI needed these resolutions to resolve features within the design spaces, so must a bounding classifier require similar resolution to make informed guesses concerning the boundaries of feasible design space regions.

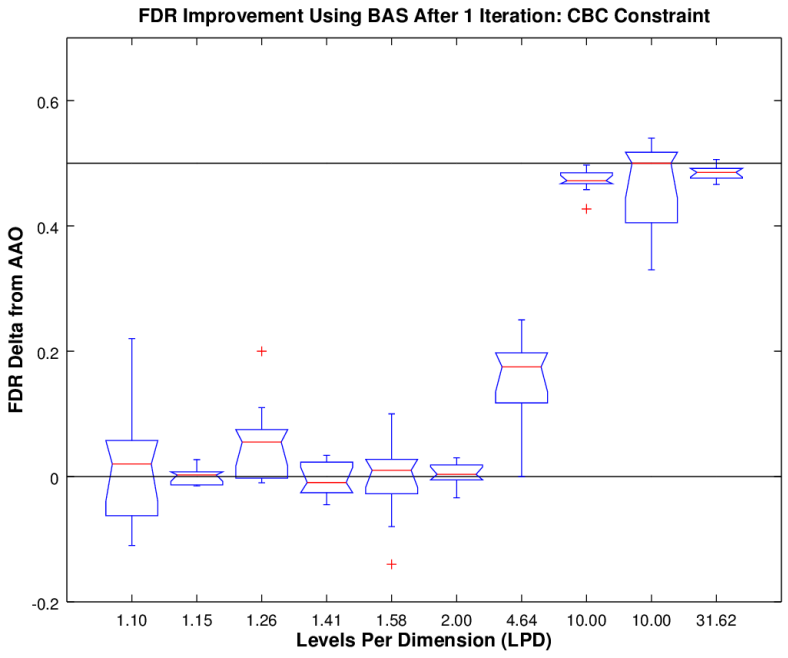

Figure 51: Effects of Bounded Adaptive Sampling for Design Spaces Subject to a Coarse Checkerboard Constraint (CBC)

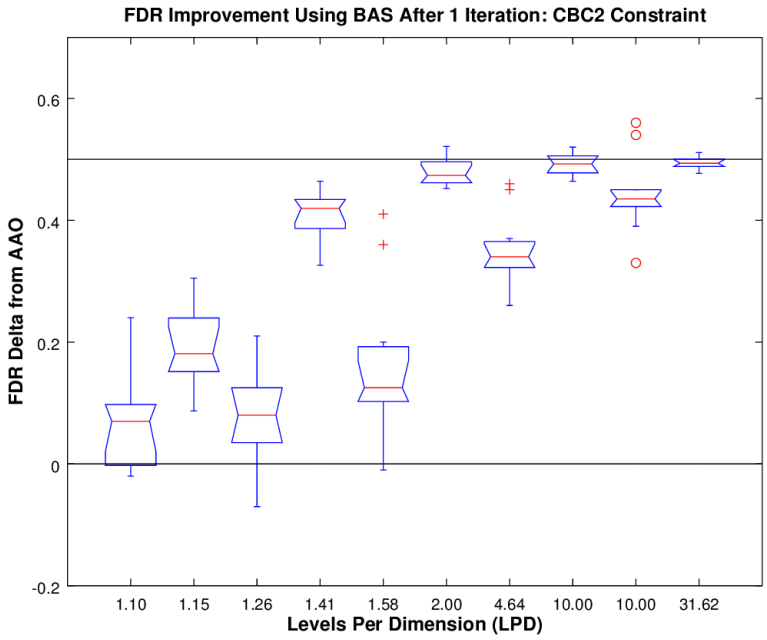

Figure 52: Effects of Bounded Adaptive Sampling for Design Spaces Subject to a Coarse Checkerboard Constraint 2-D (CBC2)

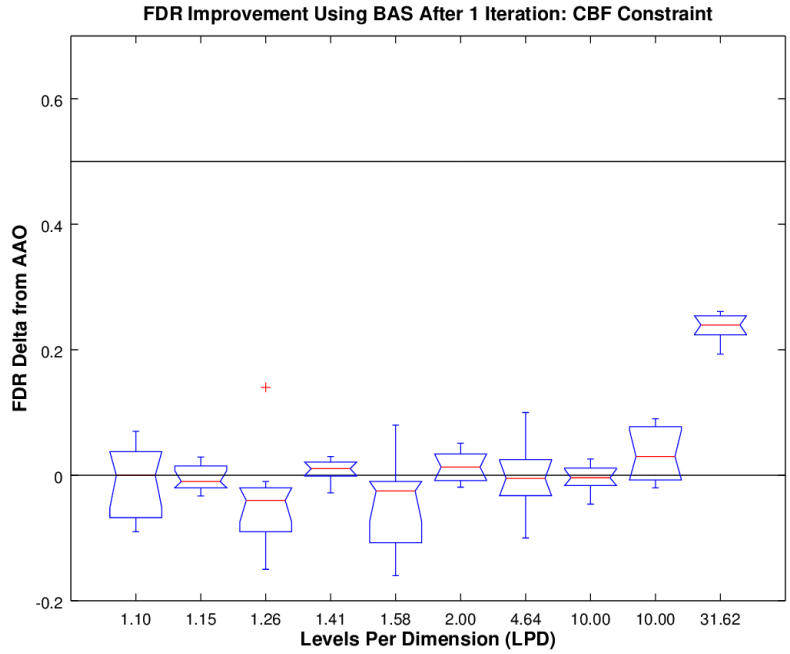

Figure 53: Effects of Bounded Adaptive Sampling for Design Spaces Subject to a Fine Checkerboard Constraint (CBF)

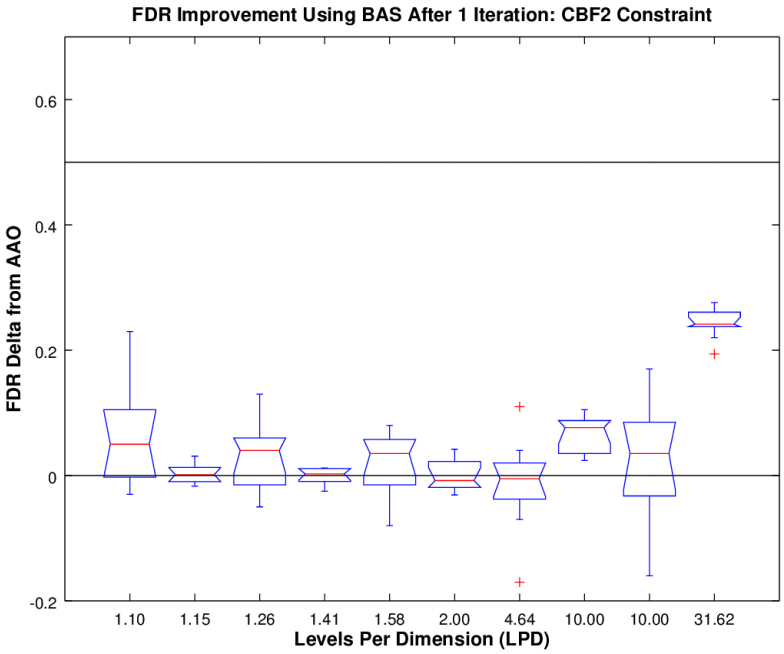

Figure 54: Effects of Bounded Adaptive Sampling for Design Spaces Subject to a Fine Checkerboard Constraint 2-D (CBF2)

5.2.5 Conclusions and Consequences

Experiment 2 was performed to evaluate the potential benefit of utilizing a bounding for a NHC design space to perform adaptive sampling and thus reduce the computational burden when exploring these spaces. This approach termed Bounded Adaptive Sampling BAS was compared against the traditional approach of sampling All At Once AAO within a bounding hypercube defined by the limits on the design variables spanning the space. Two metrics were used to measure and compare the performance of these two methods: Method Execution Time MET and the Feasible Design Ratio FDR. While the MET difference was found to be effectively negligible for the design spaces evaluated and bound, examining the difference in FDR between the two approaches yielded interesting results.

The FDR was computed for unique NHC design spaces defined by a test matrix consisting of multiple PMC DOEs defined by differing numbers of dimensions and cases per design space. FDR was computed for the AAO cases and then for the new samples produced by the BAS approach. To suggest these new samples the BAS approach constructed random forest classifiers from initial sample sets in each of the constrained design spaces. These classifiers were then queried to produce a set of new samples they believed would be feasible.

It was initially hypothesized that using any kind of bounding at all and hence leveraging information about the design space would lead to a better rate of return when re-sampling a design space. However, through the experiment it quickly became apparent that akin to experiment 1, the resolution at which the space was initially sampled was hugely important in determining the success of any bounding and adaptive sampling approach attempting to leverage design space information. In instances where the resolution (measured in LPD) was very low, the BAS approach performed poorly and at times worse than the traditional AAO sampling. Interestingly, similar LPD values to those seen in Table 5 were required to begin to resolve features within

the design space and begin illustrating a positive benefit associated with performing BAS over AAO sampling.

Additionally, the experiment revealed that a higher dimensional space, with only a subset of its dimensions being NHC, can be seen as effectively collapsed from the point of view of a bounding classifier for the NHC regions. This means that when constructing a bounding for the NHC design space LPD is artificially increased due to the collapse of Hypercubic dimensions. While these dimensions do not disappear, they are simply irrelevant for the NHC bounding and thus the design space (for BAS purposes) can be thought to collapse to a lower dimensional space in only the NHC variables. This powerful result has the potential to allow high dimensional NHC design spaces (where only a subset of the dimensions are NHC) to be successfully bound and explored with significantly lower resolution (LPD) than the previous experimental results otherwise seem to require.

Ultimately, experiment 2 was able to show that if a bounding is constructed using sufficient resolution to resolve the features present within the NHC design space, then it will provide an advantage for future exploration of the space. Thus, hypothesis 2 was considered substantiated. With this conclusion, the next step was to determine how to best perform this BAS for practical problems of interest in which the achievement of high LPD is less likely and the construction of accurate boundings a little less straightforward.

5.3 Experiment III

5.3.1 Motivation and Thought Experiment

Experiment 2 established that a bounded feasible design space is generally helpful for future exploration, but the question remained how should this bounding be constructed? The ultimate goal is to determine an accurate representation of the global feasible design space. Experiment 3 was created to evaluate the two main options for

constructing this representation. The investigation was meant to determine whether a bouding should simply be constructed for this global space, or if it is perhaps more advantageous to use Set Based Design SBD Principles to fit individual constraints and then find a global set by integrating through intersection later.

Based upon research in SBD and results from experiment 2 which suggested improvement in the accuracy of boundings due to the collapse of high dimensional spaces into NHC sub-spaces of lower dimension, it was hypothesized that a SBD approach for computing the global feasible space would be the most efficient. This assertion is largely supported by the assumption that while a design space may be NHC, it is unlikely that the space is NHC in all of the design variables simultaneously for a real physics-based application. The situation that is assumed to be much more probable is that certain sets of design variables are related to each other through correlations or constraints which themselves are a product of the physics embedded within the problem. For example, it is much more probable to expect a fan blade parameter to be correlated with a turbine blade parameter as opposed to a wing design variable. Based upon this logic, hypothesis 3 is expressed formally as follows:

5.3.1.1 Hypothesis 3: Regarding RQ3

If a set-based design (SBD) approach, which integrates through intersection multiple Constraint Defined Feasible Sets (CDFS), is used to construct a global boundary of the feasible design space, then this approach will provide a more efficient and accurate representation of the true feasible space than simply bounding the global feasible set.

5.3.2 Experiment Design

In order to investigate whether a SBD approach would allow for the efficient and accurate creation of a bounding for the global feasible space, a 10 dimensional NHC design space subject to 4 separate constraints was examined. For the hypothesis presented in this experiment to be substantiated, the SBD approach must provide a

more accurate and complete representation of the global feasible design space than the global bounding approach through the expenditure of comparable computational resources. PMC sampling was used to populate the 10 dimensional baseline design space which was then made NHC by the simultaneous application of 4 separate constraints, each a function of different subsets of the 10 design variables. Both bounding methods were then examined for their ability to accurately capture the global feasible space within this constrained design space.

5.3.2.1 Apparatus

To determine if a SBD approach was superior to a global approach for BAS of NHC spaces, a representative NHC test design space subject to multiple unique constraints was required. This test design space would be created through NHC constraints simultaneously applied numerically (See Appendix for details) to the baseline Hypercubic design space defined for each repetition of the design space sample. The constraints were applied on different subsets of the DV considered within the test design space. These constraints would be queried again once adaptive samples had been generated with the use of the bounding classifiers to determine which designs among the adaptive samples were feasible with respect to each relevant constraint.

As this experiment was meant to determine the most appropriate approach for constructing a bounding of the global feasible design space for adaptive sampling purposes, a means for constructing boundings was again necessary. Even though the primary test in this experiment examined the difference between two approaches for the construction of a global bounding, it also examined two competing classifier tools. The randomForest package implemented in the statistical programming language R was again utilized to construct random forest *RF* classifiers for the purpose of bounding feasible design spaces for this experiment [83, 23]. RF classifiers were further used in this experiment to identify NHC variables relevant to each unique

feasible set within the design space. In Addition, the kernlab package implemented in R was used to construct kernel-based support vector machine SVM classifiers to provide an alternative bounding tool to the RFs [83, 3]. The parameters used to construct the RF and SVM classifiers are as follows:

- **Random Forest** RF: Number of trees = 2000
- **Support Vector Machine** SVM: kernel: Radial Basis Function RBF, σ = 0.1, Cost = 10

5.3.2.2 Metrics

Three individual metrics were utilized to evaluate the efficacy of the two bounding methods being compared. They were meant to capture the accuracy of the boundings created by the methods as well as quantify the associated computational costs to construct them. The three metrics are as follows:

- **Method Execution Time** MET: the computational time (measured in seconds) required for the respective methods to generate their required elements. For the global bounding approach, this includes the time required to create the global bounding in all NHC dimensions. For the SBD approach, this represents the time required to create each of the CDFS boundings and assemble the integrated bounding for the global set.
- **Feasible Design Ratio** FDR: the ratio of feasible designs obtained within the sample set to the total number of designs evaluated within the sample set. This metric measures how successful each bounding method is at suggesting feasible designs. FDR is decreased from a upper limit value of unity due to the Number of False Positives NFP present within a new sample suggested by the bounding. False Positive designs are designs which are incorrectly classified as

feasible by the bounding, but are actually infeasible when evaluated with the truth model.

- **Number of False Negatives** NFN: the opposite of False Positives, the NFN represents the total number of designs which are incorrectly classified as infeasible by the bounding (and thus not suggested for a new adaptive sample), but are actually feasible when evaluated with the truth model. The NFN can be seen as a measure of how conservative a particular bounding may be, with a large NFN value suggesting a conservative bounding which potentially excludes a large region of the true feasible space.

5.3.3 Experiment Settings and Execution

This experiment required the comparison of two competing approaches performing bounded adaptive sampling for a Non-Hypercubic design space. To simplify the extent of this experiment, only a single NHC design space was examined over 10 repetitions initially sampled with 1000 design cases each. This design space featured 10 total design variables of which 8 were made NHC by the imposition of four separate constraints. The constraints applied to the design space and the variables they affected are as follows:

- **Non-Linear Constraint Large (NLCL)**: 3-D; Dimensions Affected: X1, X3, X5; Infeasible Volume: 20 percent of baseline hypervolume
- **Linear Constraint Small (LCS)**: 2-D; Dimensions Affected: X2, X4; Infeasible Volume: 5 percent of baseline hypervolume
- **CheckerBoard Constraint Coarse (CBC)**: 2-D; Dimensions Affected: X5, X7; Infeasible Volume: 50 percent of baseline hypervolume
- **Hyperspehere Removal Constraint (HS)**: 3-D; Dimensions Affected: X4, X8, X10; Infeasible Volume: 20 percent of baseline hypervolume

These particular NHC subsets were chosen to help illustrate the potential utility of the SBD bounding approach as well as allow for visualization of the NHC regions of the 10-D design space. The global feasible space for this 10-D design space is simply the intersection of the feasible regions of these four Constraint Defined Feasible Sets (CDFS). It is important to note that while the design space being examined is NHC in 8 of its 10 design variables, only small subsets of the 10 variables are relevant to each CDFS. Thus, in the SBD bounding approach, boundings for each of the individual CDFS only need be constructed as a function of the variables relevant to that particular set (the rest are assumed Hypercubic for that set and can be sampled as usual by picking a value between design variable limits). This property, combined with the artificial increase in LPD due to the collapse of Hypercubic dimensions, highlights the logic behind hypothesis 3.

Before the procedure of experiment 3 is enumerated in detail, it is necessary to call attention to two vital assumptions being made. Firstly, in order for the SBD bounding approach to be distinct from the global bounding approach, there must be output with sufficient detail from an initial sample of the space to identify the presence of multiple modes of infeasibility (i.e. there must exist more than one CDFS) within the NHC design space. If this is not the case, then only a single CDFS can be constructed from the data and this is by definition the global feasible set. Because of this distinction, it is hypothesized that the more modes of infeasibility a given design space has, the more potentially advantageous it is to perform bounded adaptive sampled through the use of the SBD approach. The second assumption being made, and critical to the potential advantages of SBD bounding, is that the variables are relevant to the boundary of each CDFS are able to be identified. If these variables are not able to be extracted from the full set of variables composing the design space then boundings must be constructed as a function of all variables and thus do not benefit from the artificial increase in LPD that occurs due to the collapse of Hypercubic dimensions.

For the purposes of this experiment, the Gini ranking of variables (features) provided by a random forest classifier fit to the global feasible space in all 10 dimensions was used to identify and separate the NHC and Hypercubic variables. Furthermore, another random forest was fit to each CDFS in only the NHC to further isolate the subset of NHC variables relevant to that particular CDFS. The SBD approach then used the appropriate variable subsets from which to construct boundings for each CDFS.

5.3.3.1 Procedure

To ultimately compare the two approaches, the mean values of the metrics over all 10 replications were used to determine the superior method for bounded adaptive sampling for the given design space. The enumerated procedure followed through the experiment is as follows:

1. Construct a PMC design in 10-D with 1000 cases and impose certain constraints on different subsets of variables meant to illustrate different characteristics potentially present in a NHC design space, make 10 repetitions of each of this space

2. Classify designs as either successes or failures w.r.t each constraint

3. Classify designs as either global successes or failures w.r.t. all constraints simultaneously

4. Construct a global RF classifier in all DV for the purposes of variable identification

5. Taking only the variables identified as NHC with the global RF bounding, now construct global boundings (RF and SVM for comparison) in this subset of variables

6. Taking only the variables identified as NHC with the global RF bounding, now construct RF classifiers for each constraint and identify which variables appear to be NHC within this subset of varibles for each constraint

7. Taking only the subsets of variables identified as NHC for the respective constraints, now construct SBD boundings (RF and SVM for comparison) in these subsets of variables for each constraint

8. Use all of these boundings to suggest new DOEs consisting of 10000 designs each

9. Calculate performance metrics for the two approaches: predictive accuracy (FDR, NFP, NFN), timing (MET), and quality of information extracted

10. Repeat steps 2-8 for each repetition

11. Draw final conclusions

In order to draw final conclusions in this experiment, the FDR, NFP and NFN values for the new samples produced from each method were compared. MET values were also compared between approaches as a measure of computational efficiency. The hypothesis was considered substantiated if SBD consistently produced a more accurate bounding (higher FDR, and lower combined NFP and NFN) of the global feasible space than the global bounding approach utilizing comparable computational efficiency.

5.3.4 Results Discussion

To construct the input for the bounded adaptive sampling approaches, 10 replicates of the 10-D design space were generated and the four separate constraints applied to each replicate to make the spaces NHC. These replicates were then classified by determining if each case was either feasible or infeasible to each of the constraints.

A case was considered globally feasible if and only if it was feasible for all individual constraints.

With the input data generated and classified, the first examination of the space came through the generation of a RF trained to the globally feasible set utilizing all of the design variables as factors. Figure 55 illustrates the ranked variable importance produced by the RF for all of the variables. These plots can be interpreted as a 'sensitivity' of the RF classifier to the exclusion of any one of these variables from the model. Immediately evident is that variables X5 and X7 are deemed extremely important for providing an accurate classification for the global feasible space. This makes sense as these two variables were utilized in the CBC constraint which was responsible for producing the largest infeasible volume within the design space. Furthemore, it is interesting to note that X5 is ranked as slightly more important than X7 and this again makes sense as it was also utilized for the NLCL constraint. The next most important variables to the classifier are in order (referring to Gini plot): X4, X10, X8, X3 and X1. These variables are also utilized for the constraints with X4 being shared by two constraints. The two variables in which the space remained Hypercubic, X6 and X9 are the lowest ranked variables in terms of importance to the classifier, yet it is interesting that X2 is only seen as slightly more important than these variables. From this information it is difficult to discern whether or not the space is NHC in X2. This is perhaps because the constraint to which X2 is relevant is responsible for creating an infeasible region containing only 5 percent of the original Hypercubic volume. Additionally, X4 with which X2 shares the LCS constraint is already highly ranked, thus the model may be incorrectly attributing some of X2's effect on the feasible space to X4. Ultimately while providing a very useful look into the characteristics of the design space, the global RF classifier does not provide a definitive identification of the NHC variables within the design space.

After constructing the Global RF for NHC variable identification, a RF classifier

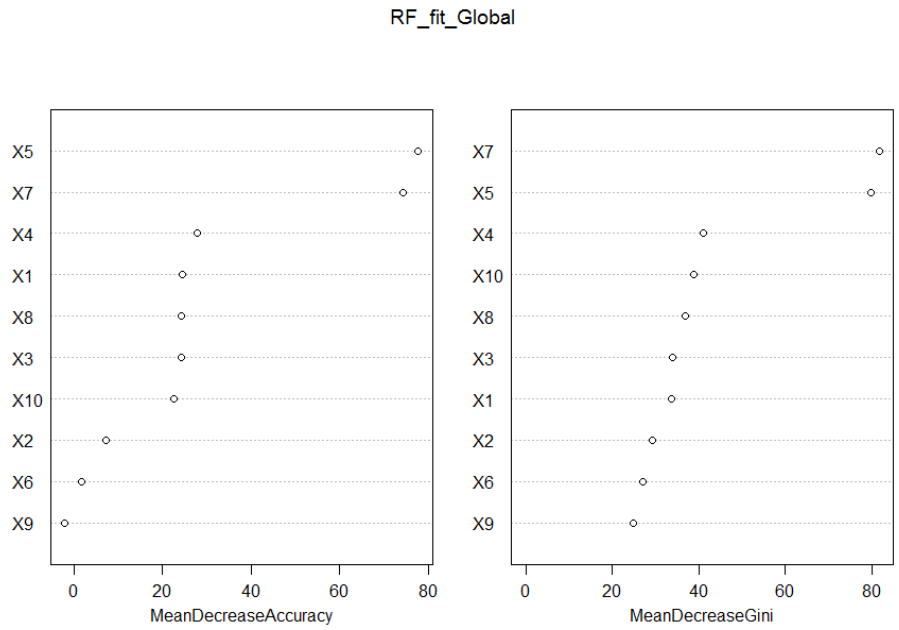

Figure 55: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding the Global NHC Feasible Space

was then trained utilizing all of the design variables as factors but classified by each individual CDFS. Figure 56 illustrates the variable importance rankings for each of these four RF classifiers. With each classifier allowed to focus on only a single CDFS, the NHC variable identification results are much less ambiguous and correctly identify the relevant set of variables for each CDFS. This result provided the first evidence that the SBD bounding approach could provide superior information about the design space compared to the global bounding approach.

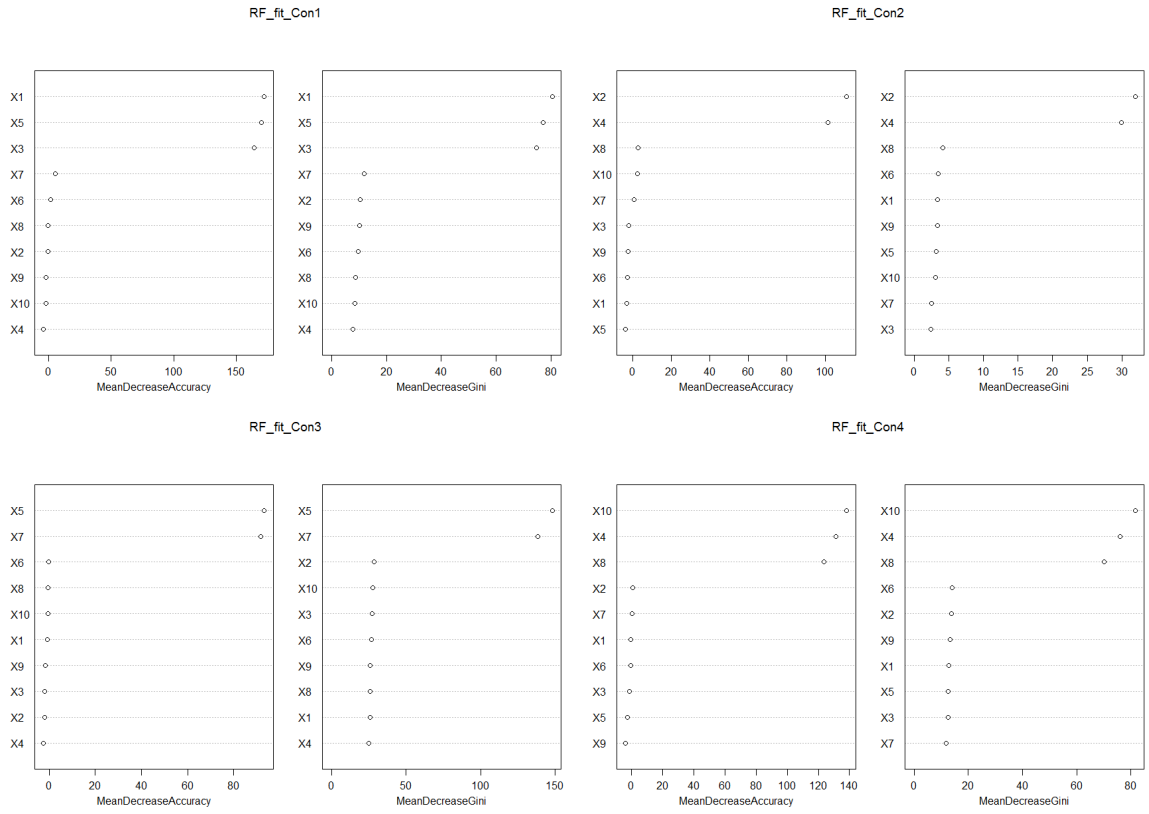

Figure 56: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding each NHC CDFS (Con1: NLCL 3-D, Con2: LCS 2-D, Con3: CBC 2-D, Con4: HS 3-D)

Taking the results from the variable identification steps, the set of variables considered for the boundings was paired down from the full 10 to 8 relevant variables: X1, X2, X3, X4, X5, X7, X8, and X10. All 8 of these variables were then utilized to construct the global boundings for the feasible design space through training both RF and SVM classifiers. For the SBD approach however, only the relevant subsets of these variables, identified with the individual CDFS trained RFs, were used to construct boundings through training both RF and SVM classifiers for each respective CDFS.

Results Interpretation (Fig. 57-60) These figures depict the new adaptive samples (at 10000 design cases each) produced by the two bounding methods (utilizing two types of classifiers each) displayed in the dimensional subsets that were relevant to each CDFS. The black circles represent design cases which were deemed feasible by the particular classifier used to bound the CDFS. Conversely, the voids (or regions of greatly reduced density) present in these samples represent regions of infeasibility identified by the particular classifier and thus the new adaptive samples are rarely present in these regions. These visualizations of cross-sections of the feasible constrained design space can be compared to their respective constrained design spaces in the Appendix to provide an indication of how well the feasible design space is being classified and bound.

It is immediately apparent through visual inspection of these adaptive samples that while both BAS methods appear to capture some structure within the NHC design space, the SBD generated samples provide more accurate representations of each of the CDFS present within the 10-D design space. This result is a direct effect of the artificial increase in LPD which occurs as dimensions not relevant to a particular set are ignored/collapsed when boundings are constructed through the SBD approach. To quantify this, effective LPD can be calculated for both the global bounding and

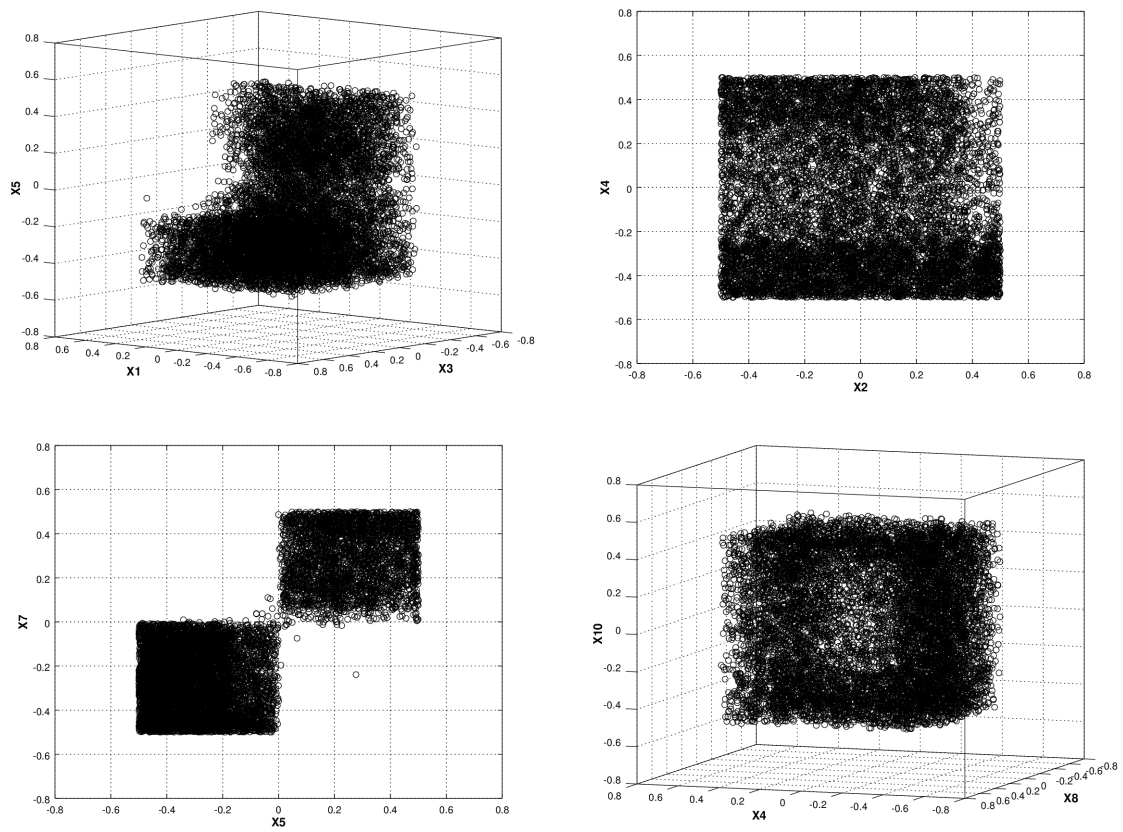

Figure 57: New Designs Suggested by the Global Bounding Approach for each of the CDFS (NLCL 3-D, LCS 2-D, CBC 2-D, HS 3-D) using a Random Forest (RF) Classifier

SBD bounding approaches:

$$LPD_{Global} = n^{\frac{1}{d_{relevant}}} = 1000^{\frac{1}{8}} = 2.371 \tag{31}$$

$$LPD_{SBD-3D} = n^{\frac{1}{d_{relevant}}} = 1000^{\frac{1}{3}} = 10.000 \tag{32}$$

$$LPD_{SBD-2D} = n^{\frac{1}{d_{relevant}}} = 1000^{\frac{1}{2}} = 31.623 \tag{33}$$

As evidenced by the calculations, the resolution available for training classifiers using the global bounding approach is an effective LPD of 2.371 as the bounding is fit to all 8 NHC dimensions. Using the same set of training data, the equivalent resolution (expressed in LPD) is increased by a factor larger than 4 for the 3-D CDFS boundings and by a factor greater than 13 for the 2-D CDFS boundings constructed with the SBD

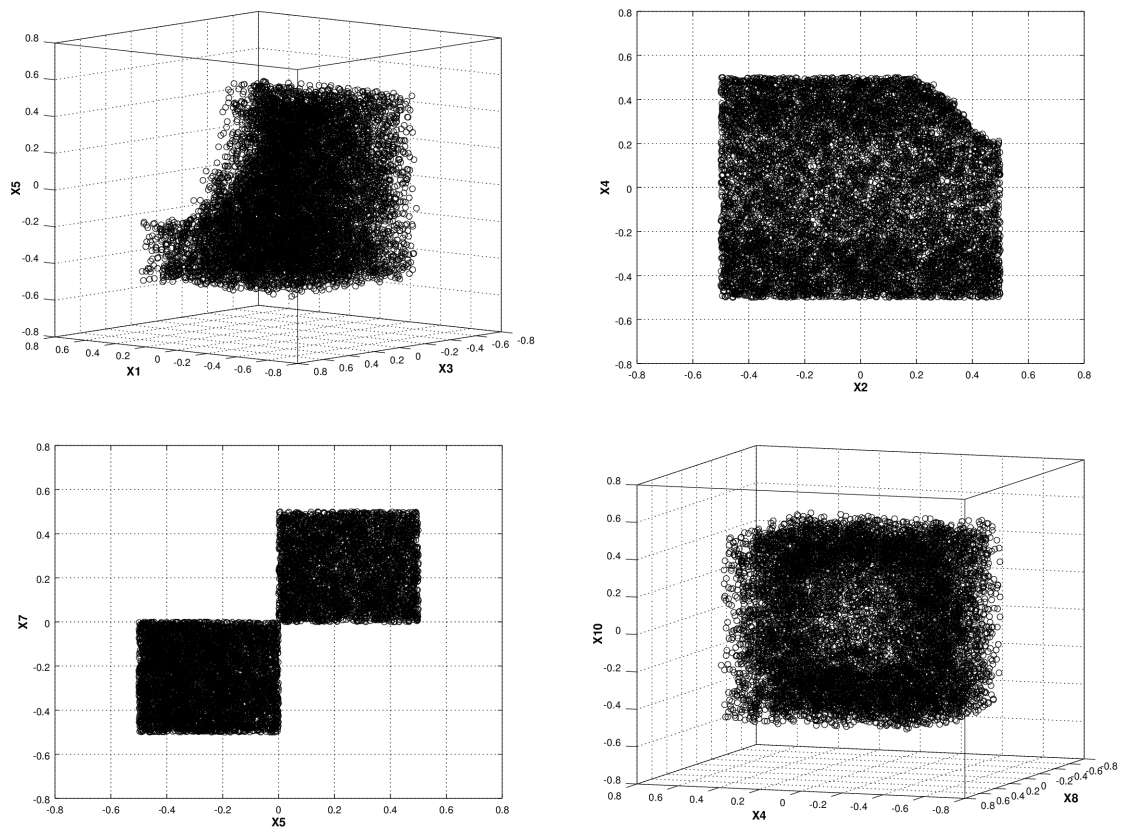

Figure 58: New Designs Suggested by the Set-Based Design (SBD) Bounding Approach for each of the CDFS (NLCL 3-D, LCS 2-D, CBC 2-D, HS 3-D) using a Random Forest (RF) Classifier

approach. Not only does this strongly suggest that the SBD constructed boundings will be more accurate, but by breaking out the individual CDFSs, more information is returned about the design space. This realization is important not only because it provides a greater granularity of insight into the features of the feasible design space (useful for debugging codes, verification) but also because it provides a more robust bounding of the design space. If any one of the constraints currently present in the design space were to become inactive, a new bounding for the feasible space would need to be generated if only a global bounding exists. However, if the SBD approach is used, the new global feasible space is simply the intersection of the remaining CDFS boundings and thus no new bounding classifier need be trained.

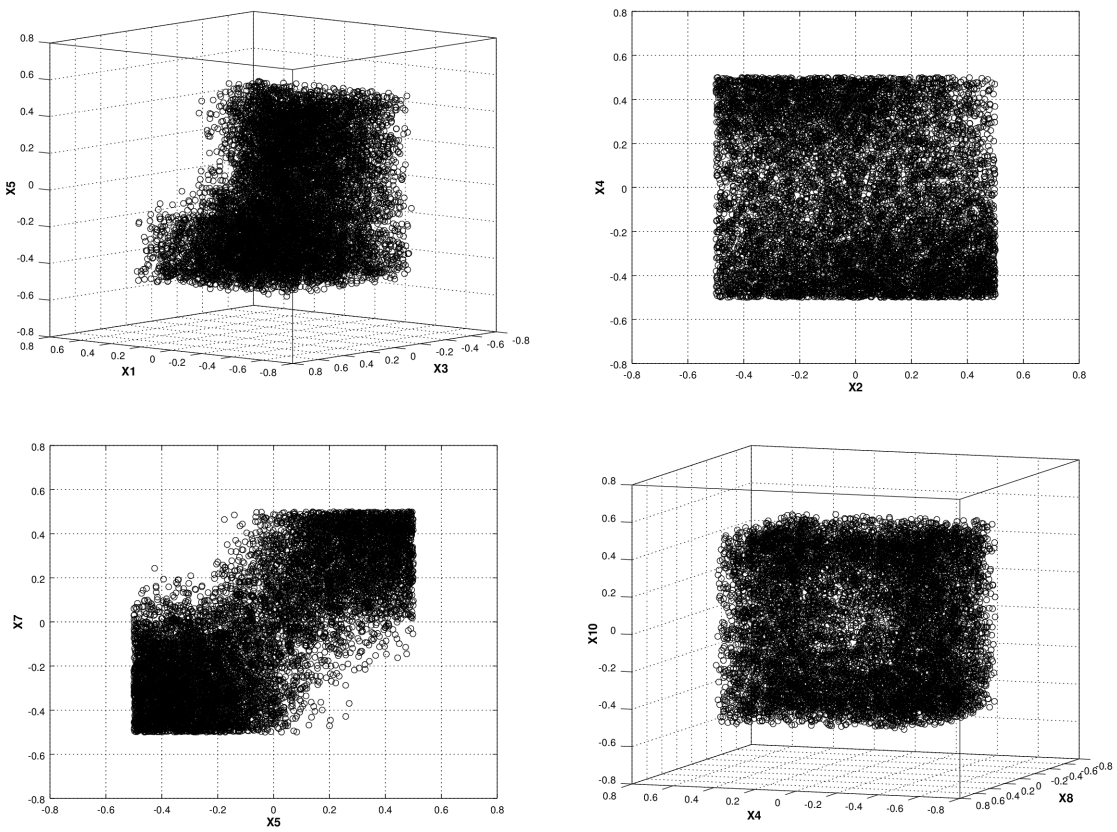

Figure 59: New Designs Suggested by the Global Bounding Approach for each of the CDFS (NLCL 3-D, LCS 2-D, CBC 2-D, HS 3-D) using a Support Vector Machine (SVM) Classifier

While the SBD approach certainly seems to produce adaptive samples which most accurately represent the feasible design space delineated by each of the constraints, it is interesting to observe how the different classifier tools perform. Through visual inspection of these samples, it is difficult to discern if the RF classifiers provide better boundings than those constructed from SVM classifiers. It appears that while the RF classifiers are able to much more accurately capture the features present due to the CBC constraint, the SVM classifiers perform better at capturing the LCS constraint. It is also difficult to determine which classifier type better resolves the non-linear features. Based on this result, both of these classifier tools will be utilized in the final experiment.

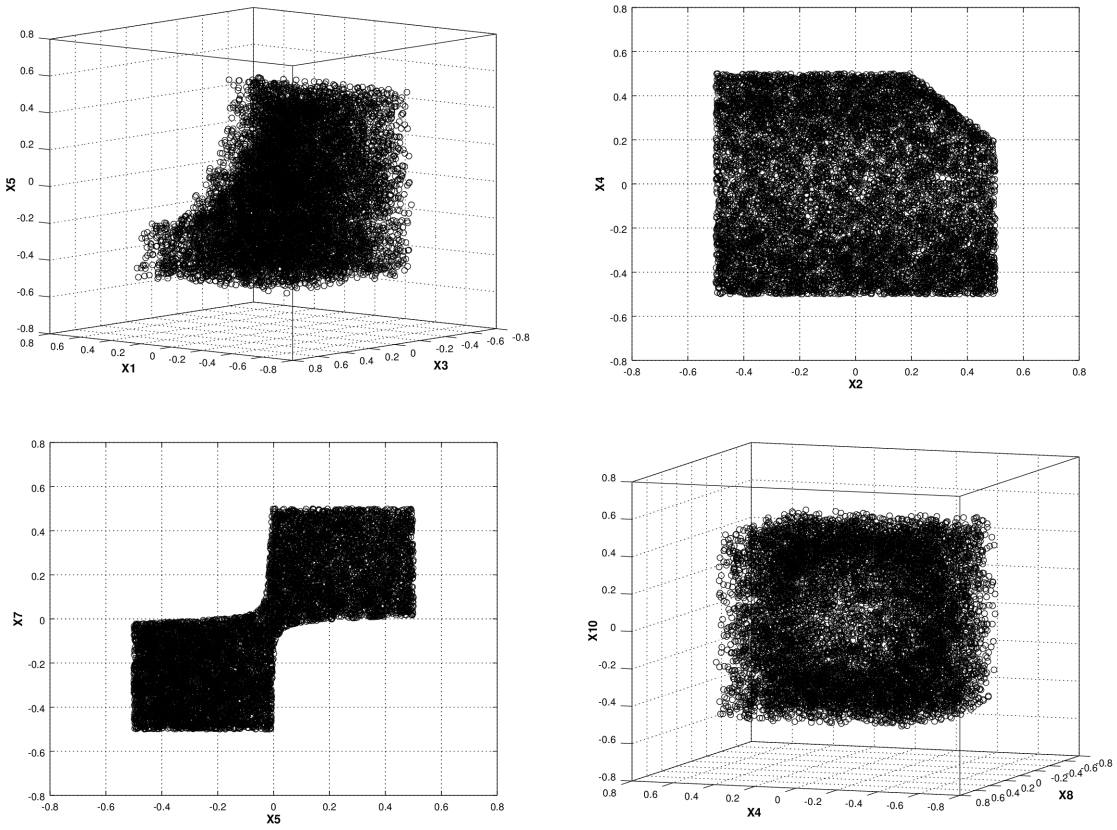

Figure 60: New Designs Suggested by the Set-Based Design (SBD) Bounding Approach for each of the CDFS (NLCL 3-D, LCS 2-D, CBC 2-D, HS 3-D) using a Support Vector Machine (SVM) Classifier

More concrete conclusions can be drawn for the two competing BAS approaches through examination of the quantitative results in each of the metrics. Table 7 displays the MET results obtained through the experiment. While the SBD approach is more computationally expensive than the global approach in all cases, it is only slightly so for this particular design space and its associated CDFS. It is expected that the MET would increase significantly with the size of the design space in both approaches, however depending on the number of CDFS to be evaluated in the SBD case this could lead to significant differences in computational effort between the two approaches. However, it is worth noting that although the SBD approach required more total MET, it appears that the required time to construct each individual bounding was noticeably less than the time to construct the global bounding. This result makes sense as the SBD boundings were constructed in fewer variables than the global bounding. When comparing the use of RF to SVM classifiers, although RF required more resources, for this particular design space, the difference would be considered negligible in the decision to choose one over the other. To provide a hardware benchmark, these tests were performed on a system running 64-bit Windows 10 with an Intel Core i7-4790k processor at 4.00 GHz with 16 GB of RAM.

Table 7: Summary of Method Execution Time (MET) Required for AAO and BAS

		Training of Bounding Classifier	
Bounding Approach	Classifier Type	min time (s)	max time (s)
Global	Random Forest	2.130	2.47
Set-Based Design	Random Forest	3.210	3.490
Global	Support Vector Machine	0.160	0.220
Set-Based Design	Support Vector Machine	0.220	0.310

To evaluate the accuracy of the competing BAS approaches they were compared using the FDR, NFP and NFN metrics. Figures 61-65 illustrate the FDR of the adaptive samples for each of the individual CDFS and finally the global feasible space

generated by the bounding classifiers produced through both approaches (Global and SBD) using the two classification tools (RF and SVM). As there were 10,000 new design cases suggested per adaptive sample set, the FDR for each replication was calculated as follows:

$$FDR = \frac{10000 - NFP}{10000} \tag{34}$$

Through this relation the FDR captures the NFP generated by a given bounding produced adaptive sample and scales it to easily allow accuracy conclusions to be drawn. For every CDFS, FDR technically ranges from 0 to 1, but an FDR below the fraction of the original Hypercubic volume bound by a given CDFS would mean that the BAS method performed worse than simply randomly sampling the original Hypercubic space. The beneficial lower limits of FDR for each of the CDFS are the following:

- **NLCL FDR Lower Limit**: 0.80
- **LCS FDR Lower Limit**: 0.95
- **CBC FDR Lower Limit**: 0.5
- **HS FDR Lower Limit**: 0.80
- **ALL (Integrated CDFS) FDR Lower Limit**: 0.30

Analyzing the FDR results in Figures 61-65 a few general trends emerge. As expected, the SBD boundings result in a larger FDR with generally less spread between repetitions than their globally bound counterparts. It is surprising then to see when all of the CDFS are integrated together to estimate feasible cases for the global feasible space that the globally bound RF classifier has such a high FDR. This bounding was constructed with much less resolution yet it only differs in mean FDR from the SBD RF by less than 0.025. This result suggests that the Global RF bounding is perhaps much more accurate than expected and maybe the SBD approach while slightly

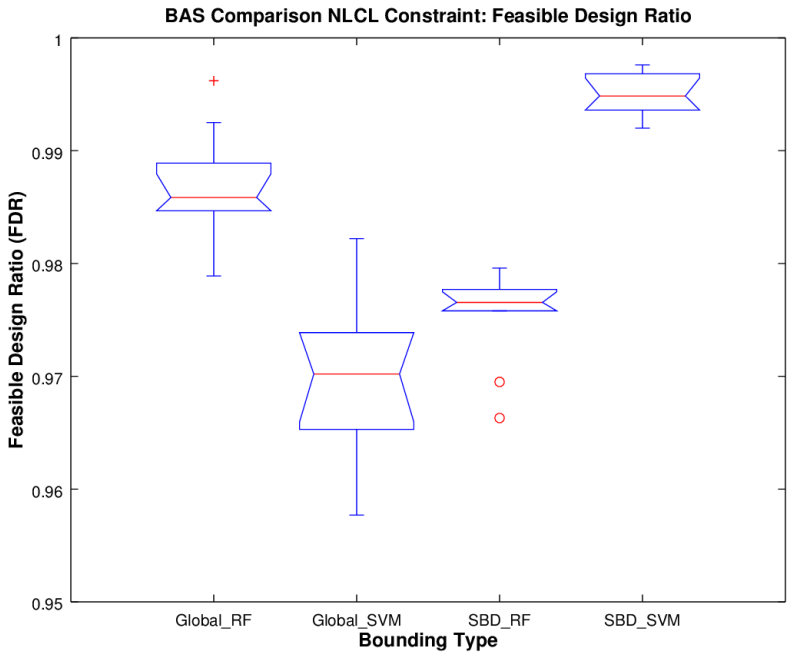

Figure 61: Feasible Design Ratio (FDR) Comparison Between Bounding Approaches for the Large Non-Linear (NLCL) Constraint Defined Feasible Set (CDFS)

more accurate may not be worth the additional effort. However, this is because FDR (and its surrogate NFP) alone do not tell the entire story regarding accuracy.

Figure 67 shows the other side to the accuracy story and illustrates that while the Global RF bounding may have a high FDR, it also has an extremely high number of false negatives. While the false negatives are not infeasible cases, they represent feasible designs passed up by the classifier because it was too conservative in its bounding of the feasible space. This high NFN number means the Global RF bounding passed on almost double the number of truly feasible case for its adaptive sample than any of the other boundings. Thus, when looking at all the metrics it is clear that SBD BAS approach provides a clear advantage to global BAS without significantly higher computational effort required.

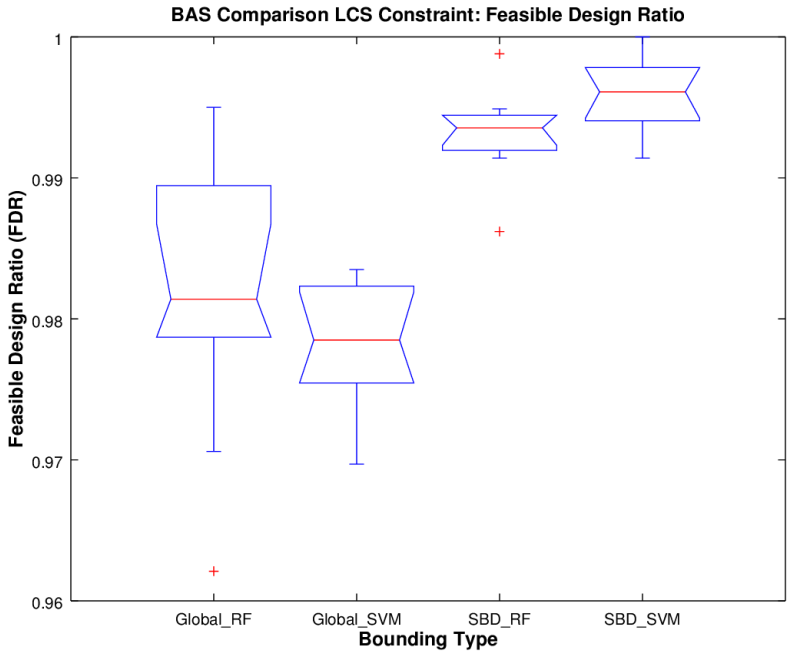

Figure 62: Feasible Design Ratio (FDR) Comparison Between Bounding Approaches for the Small Linear (LCS) Constraint Defined Feasible Set (CDFS)

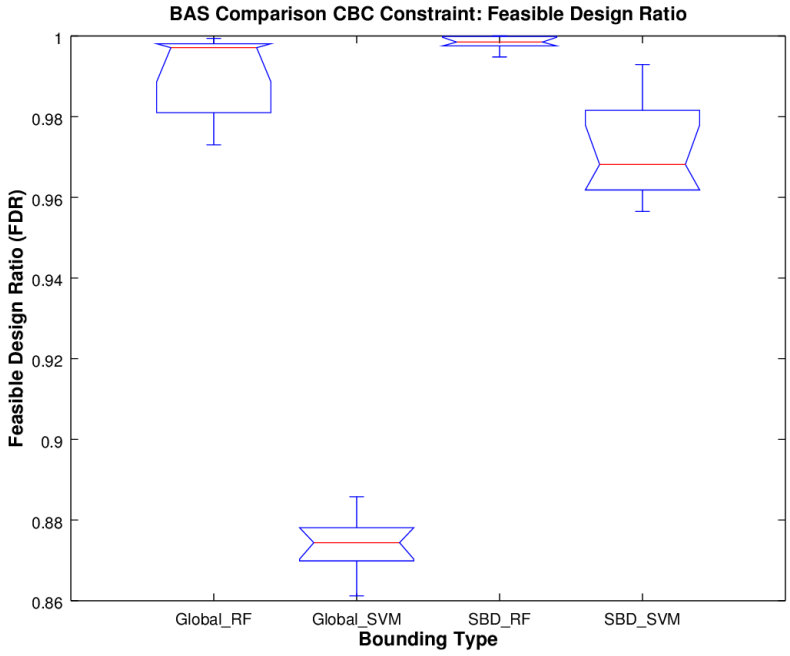

Figure 63: Feasible Design Ratio (FDR) Comparison Between Bounding Approaches for the Coarse Checkerboard (CBC) Constraint Defined Feasible Set (CDFS)

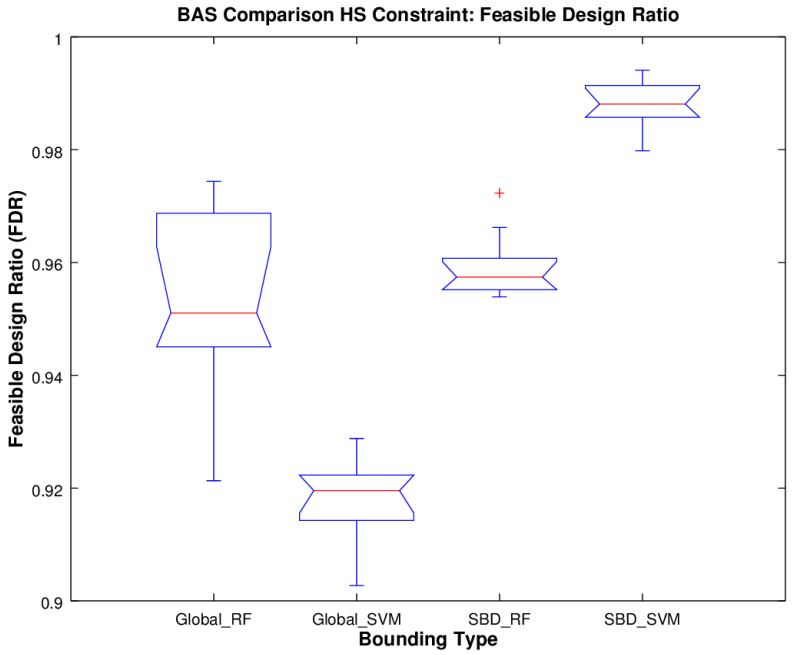

Figure 64: Feasible Design Ratio (FDR) Comparison Between Bounding Approaches for the Hypersphere Removal (HS) Constraint Defined Feasible Set (CDFS)

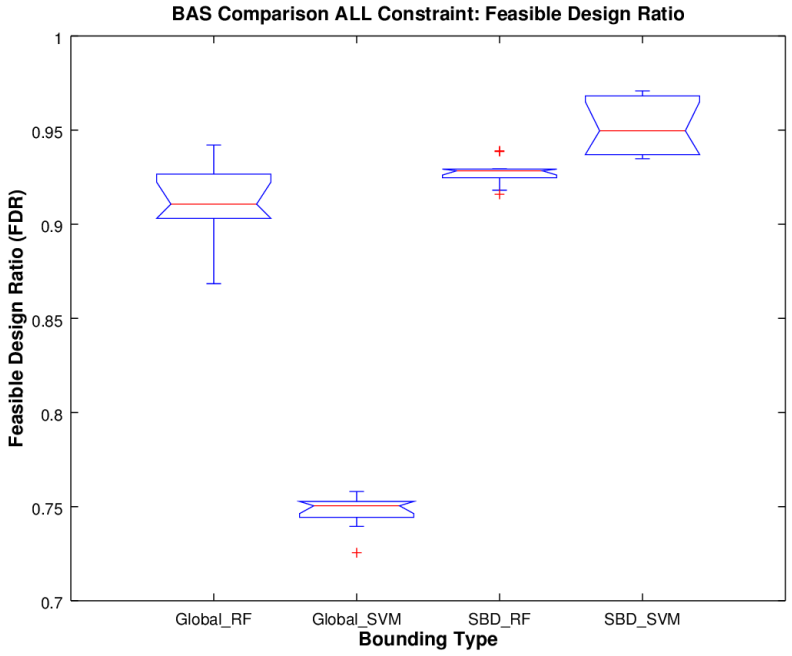

Figure 65: Feasible Design Ratio (FDR) Comparison Between Bounding Approaches for the Global Feasible Set (Intersection of All CDFS)

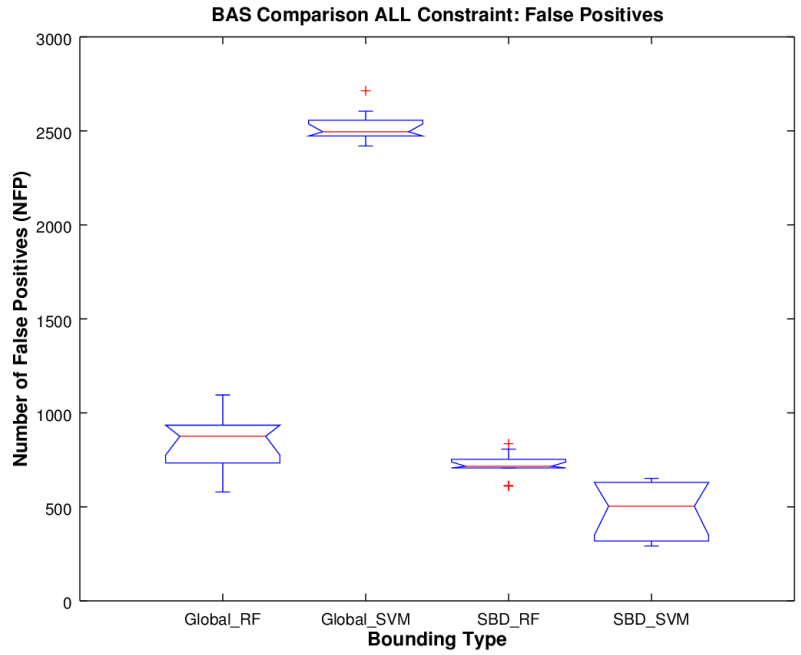

Figure 66: Number of False Positives (NFP) Comparison Between Bounding Approaches for the Global Feasible Set (Intersection of All CDFS)

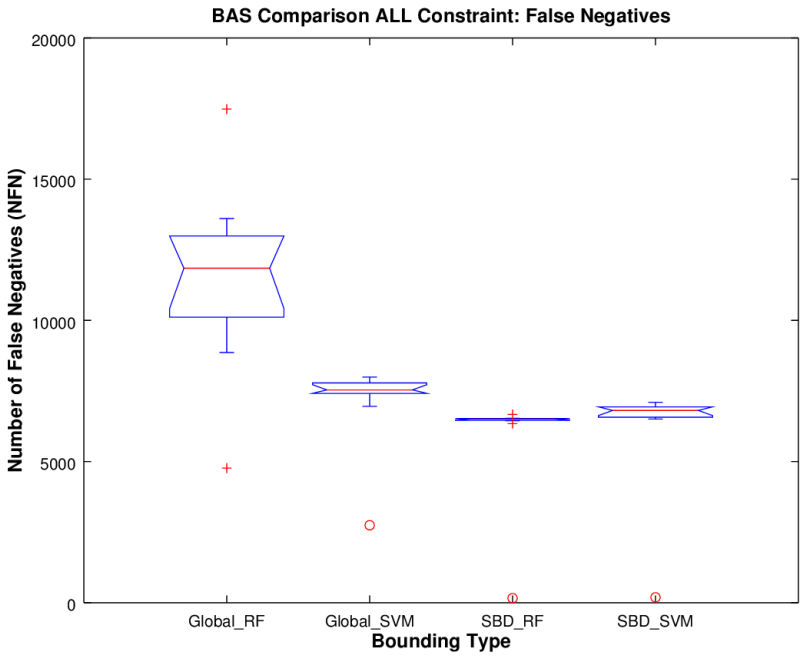

Figure 67: Number of False Negatives (NFN) Comparison Between Bounding Approaches for the Global Feasible Set (Intersection of All CDFS)

5.3.5 Conclusions and Consequences

Through analysis of the results of experiment 3 it was established that although the SBD approach for bounded adaptive sampling required slightly more computational effort, it was generally more accurate than a global approach. If the feasible regions formed by the CDFS are disjoint, then from the same set of training data a smaller subset of variables can be identified that are relevant to the given CDFS and thus the other dimensions while not ingnored can effectively be collapsed. This collapsing of dimensions artificially increases the resolution (LPD) used to generate the bounding for that particular CDFS through the reduction of the number of variables or factors used to construct the bounding classifier. This results in SBD boundings that are more accurate (higher FDR and lower NPF and NFN) and also benefit from not needing to fit the complex composite structure likely present in the global case due to its definition as the intersection of all CDFS. Based upon these results hypothesis 3 is considered substantiated, however care should be taken when examining spaces of high dimension with large numbers of CDFS as these factors could drive the computational cost of such an approach significantly higher.

5.4 Experiment IV

5.4.1 Motivation and Thought Experiment

The purpose of experiment 4 was to demonstrate the methodology proposed within this thesis on a real aircraft conceptual design problem using a state-of-the-art Modeling and Simulation environment to produce data. The key theme of this final experiment was to assemble all of the knowledge gained in the previous experiments and utilize the resultant complete methodology to show an improvement in DSE efficiency and knowledge gain with respect to current practices. If such a benefit can be shown, this would provide evidence of the effective utility, capabilities and likely limitations of using such an approach for Hypercubic classification and DSE for practical

problems. To this end, research question 4 is reiterated:

> **Research Question 4 (RQ4)**:
>
> Can use of the proposed methodology demonstrate an improvement in efficiency and knowledge gain with respect to state of the art practices in design space exploration techniques for a realistic aircraft conceptual design problem?

In order to answer RQ4, the complete methodology was put through its paces for a high-dimensional aircraft design problem which utilized the Environmental Design Space (EDS) to explore the conceptual design space for a Large Twin Aisle (LTA) Hybrid Wing-Body (HWB) aircraft concept. EDS was chosen as the Modeling and Simulation environment for this particular problem due to its widespread use for conceptual studies [17, 30, 32, 38, 40, 39, 43, 44, 65, 73, 74, 75, 77, 76, 87, 88, 89], the relative expense associated with DSE within the environment, and the observed presence of multiple failure modes which appear non-random in nature. The LTA HWB was chosen as an aircraft concept for investigation because a useful and relevant design space could be expressed in less than 100 variables yet due to the advanced nature of the concept, a certain percentage of infeasible/failed designs were expected within the design space of interest. The hypothesis that the methodology is beneficial for DSE for such a conceptual aircraft design problem is considered substantiated if and only if its application allows for more efficient exploration and understanding of the design space with respect to common practices of DSE such as single iteration PMC.

5.4.1.1 Hypothesis 4: Regarding RQ4

If the design space of interest is Non-Hypercubic and sufficient design space sample resolution is utilized, then use of the proposed methodology will demonstrate an improvement in efficiency and knowledge gain with respect to state of the art practices in

design space exploration techniques for a realistic aircraft conceptual design problem.

5.4.2 Experiment Design

In order to demonstrate the methodology in action for a real problem, a 97-D design space was initially examined followed by a 50-D design space for the LTA HWB within the EDS Modeling and Simulation environment. The LTA HWB was modeled within EDS as 300 passenger (24 first class, 54 business class, 227 tourist, 2 flight crew) aircraft flying a cruise-climb mission profile with a design range of 7530 nautical miles. The design aircraft was powered by two (parametrically defined) fuselage-mounted high-bypass turbofan engines of the 50,000-60,000 pound thrust class. The vehicle had a maximum cruising altitude of 43,000 feet and a cruise Mach Number of 0.84. Additional details about the variables used to create the LTA HWB design spaces (97-D and 50-D respectively) can be found in the Appendix.

For the hypothesis presented in this experiment to be substantiated, the methodology must illustrate that it can improve upon current best practices as well as provide knowledge about the design space previously unknown or attainable only after much iterative exploration of the design space/knowledge of the underlying physics in the Modeling and Simulation environment.

The initial 97-D LTA HWB design space was a previously generated data set using EDS version 5.4. This space was explored with 15,000 unique design cases structured within the design space as follows:

- 3,000 Face Centered points (placed at extremes of the design space)
- 10,000 Space Filling points (placed randomly in the design space using PMC)
- 2,000 Random Technology Packages (designs corresponding to technology combinations, these cases only span a subset of the full design space)

These design cases would ultimately be classified and partitioned into an 'initial' set

for training consisting of the first 10,000 cases (the 3k Face Center and 7k of the PMC) and a 'validation' set containing the remaining 5,000 design cases.

The 50-D LTA HWB design space was ultimately created to address resolution and homogeneity issues that limited the successful application of the method on the initial 97-D design space. The 50-D design space used a pared down subset of the original 97 design variables, removing all noise variables and keeping a set of 50 variables considered important to the responses. Additionally, a design variable governing takeoff thrust (TO thrust) was added with aggressive ranges and the lower limit for the Thrust to Weight Ratio variable (TWR) was decreased to allow for the data set to achieve more balance between feasible and infeasible designs. This space was explored multiple times, first with 15,000 design cases placed randomly in the design space using PMC (Run 1), then with another 15,000 PMC cases for increased resolution (Run 2) and finally with two adaptive sample DOEs generated by boundings created by the methodology for the purposes of exploring the feasible design space ASE and refining the NHC boundary ASR. Similar to the 97-D design space, the design cases in Run 1 would ultimately be classified and partitioned into an 'initial' set for training consisting of the first 10,000 cases and a 'validation' set containing the remaining 5,000 design cases. Run 2 would provide additional resolution for MI calculation and be used for additional validation. ASE and ASR would be used respectively for validation of predictive capabilities and boundary refinement of the CDFS boundings.

Ultimately, the objective of experiment 4 for both of these DSE tests was to show an improvement in understanding and efficiency of the exploration of the feasible design space compared to traditional techniques. This would be achieved through application of the methodology and generation of CDFS boundings to be used for adaptive sampling. If these boundings could be shown to improve the rate of return for a given expenditure of computational effort (without being too computationally

intensive to generate themselves) they would provide evidence for successful application of the methodology on a practical problem.

5.4.2.1 Apparatus

EDS version 5.4 was utilized as the Modeling and Simulation environment for evaluating design points for the LTA HWB. To allow for the evaluation of the number of design cases necessary to explore the spaces considered, the HTCondor software was utilized for distributed computing [99] and leveraged the processing power of multiple workstations throughout the Aerospace Systems Design Laboratory where the work was performed. Microsoft Excel was utilized for DOE generation and classification of results.

AS in experiment 1, MILCA was utilized to estimate MI for the datasets. Furthermore, routines written in R, again leveraging both the randomForest and kernlab libraries, were used for classifier model creation adaptive sample generation and data post-processing.

5.4.2.2 Metrics

The metrics for experiment 4 emphasize the amount of computational resources utilized and quality of the resultant boundings for the constraint defined feasible sets (CDFS) and the ultimate global feasible space existing within the LTA HWB design space. The methodology will iteratively update boundings of the design space, thus the accuracy of these boundings will be tracked for multiple iterations and analyzed to determine if improvement (of the boundings and understanding of the design space) occurs through adaptive sampling. The metrics for experiment 4 are as follows:

- **Mutual Information Delta** MID: the difference between the baseline MI value for a given unique DOE and the MI value calculated for the cases within that DOE which remained feasible after a given constraint was applied. If the MI delta value is positive (i.e. the MI value of the constrained space is higher

than that of the baseline) then features exist within the design space that make it Non-Hypercubic. The mathematical expression for MID is given in eqn. 27.

- **Method Execution Time** MET: the computational time (measured in seconds) required for the respective computational elements of methodology to generate their required elements. For the global bounding approach, this includes the time required to create the global bounding in all NHC dimensions. For the SBD approach, this represents the time required to create each of the CDFS boundings and assemble the integrated bounding for the global set.

- **Cross Validation Error** CVE: Because of the relative expense associated with generating data for this problem, cross validation will be utilized in the training of some classifiers during the experiment. CVE will be tracked as a metric for model accuracy but will be largely used to guide feature selection (i.e. which subset of variables should be used to produce the best bounding classifiers)

- **Number of False Positives** NFP: designs which are incorrectly classified as feasible by the bounding, but are actually infeasible when evaluated with the truth model. This metric will be calculated both by performing classifications on reserved validation sets and through the evaluation of new adaptive sample sets generated from BAS.

- **Number of False Negatives** NFN: the opposite of False Positives, the NFN represents the total number of designs which are incorrectly classified as infeasible by the bounding (and thus not suggested for a new adaptive sample), but are actually feasible when evaluated with the truth model. This metric will be calculated both by performing classifications on reserved validation sets and through the evaluation of new adaptive sample sets generated from BAS.

5.4.3 Experiment Settings and Execution

5.4.3.1 Procedure

The following experimental procedure was followed for both the 97-D and 50-D DSE tests. While the 97-D test ultimately produced unsatisfactory results and was deemed not appropriate for bounded adaptive sampling, it did provide insight into when and how the methodology could fail to improve upon the results achieved through sampling using common practices like single iteration PMC. In order to demonstrate the methodology for both DSE tests, the following experimental procedure was followed:

1. Generate or utilize an existing data set spanning a design space of interest, and use the output to classify the set of globally feasible design cases

2. Partition the classified data set into an 'initial' set for training and a 'validation' set

3. Take multiple random samples of a fraction of the 'initial' data set (where this fraction corresponds to the same fraction of design cases that were classified within the set as globally feasible) the and compute the Mutual Information (MI) of these samples

4. Compute the MI of the feasible design space (using only the globally feasible design cases)

5. Compute the MID between the Feasible MI value and the baseline MI values, if MID is consistently positive and the resolution (LPD) is believed sufficient, classify this space as Non-Hypercubic (NHC)

6. If the design space was classified as NHC and the decision has been made to proceed with Bounded Adaptive Sampling (BAS) of this space, identify modes of failure or infeasiblity and classify the data into a series of Constraint Defined Feasible Sets (CDFSs) for both the 'initial' and 'validation' sets

7. Train Random Forest (RF) classifiers on the globally feasible set and each CDFS of the 'initial' set to identify NHC variables relevant to each set

8. Construct cross-validated SVM bounding classifiers for each CDFS trained using the 'initial' set and sweeps of variable subsets (determined from relevance rankings produced by the RFs)

9. Select the 'best' boundings for each CDFS utilizing Cross Validation Error (CVE) to guide feature selection

10. Use the 'validation' sets for each CDFS to determine the quality of these boundings through computing NFP and NFN

11. Construct an adaptive sample meant to explore (ASE) the global feasible space by querying the boundings of each CDFS simultaneously for designs which are classified as feasible by all boundings

12. Construct an adaptive sample meant to refine (ASR) the boundary by querying the boundings of each CDFS simultaneously for designs which are hardest to classify (classification uncertainty is high)

13. Evaluate both new adaptive samples (ASE and ASR) using the modeling and simulation environment

14. Classify the output of the ASE to determine if the percentage of infeasible designs has decreased compared to the original DOE (better than BAU?)

15. Classify the output of the ASR and utilize it to retrain/refine all CDFS boundings

16. Repeat steps 10-15 until computational resources are exhausted or satisfaction with the bounded feasible design space is achieved

17. Draw final conclusions

5.4.4 Results Discussion

The following discussion is divided into two sections corresponding to the two DSE tests that were undertaken in this experiment. The first section details the 'failed' experiment for the 97-D design space and draws conclusions as to why the methodology could not be shown to yield appreciable improvement over common sampling practices. The next section then describes the 50-D test which was created to alleviate some of the issues which plagued the first test yet still remain of significant dimensionality to represent a realistic design space. Ultimately, both tests were successful implementations of the methodology, with the first breaking it and probing its limits and the second showing how if used under the proper circumstances, a significant improvement over traditional DSE can be achieved.

5.4.4.1 Initial Investigation: 97-D LTA HWB Design Space

Experiment 4 began with consideration of only the 97-D DSE test. The data set was first classified and was found to contain nine modes of failure/infeasibity, they were as follows:

- **FLOPS-ZFW**: FLOPS Zero Fuel Weight ZFW error (vehicle sizing)
- **ANOPP**: ANOPP error (noise)
- **CONDOR**: HTCondor error (distributed computing)
- **Cum-Noise**: Cumulative Noise below threshold (noise)
- **ROC**: Rate Of Climb insufficient error (vehicle sizing)
- **MDP**: Multi-Design Point error (engine sizing)
- **Thrust-Conv**: Thrust Convergence error (engine sizing)
- **WATE**: WATE error (engine flowpath)

- **Main-Conv**: Main Design Loop Convergence error (convergence)

For a design to be considered globally feasible within the set, it had to simultaneously satisfy all of these constraints (i.e. not be made infeasible by these error modes). A CDFS would would eventually be constructed for each of these modes, but it is important to note that regions of overlap are not visible with the level of output provided. Because of this, if a design case was made infeasible due to any one failure mode, it could not be considered feasible for any of the other modes. Thus the only feasible cases which could be used to train the CDFS boundings were those that were globally feasible. The classification breakdown of the 97-D LTA HWB dataset can be seen in table 8. An important conclusion from this table is that not only is the data set extremely sparse for a 97-D design space, but it is vary unbalanged in terms of the ratio of feasible to infeasible cases. For the CDFS with the largest number of infeasible cases in its training set (FLOPS-ZFW) the feasible cases still outnumber the infeasible cases by a ratio greater than 8:1, for the next largest CDFS this ratio increases to over 63:1. This unbalanced nature of the CDFS is one of the factors that ultimately contributed to the poor performance of the methodology for this particular design space.

While results from the Hypercubic classification steps suggested the design space was NHC, the resolution (LPD) at which the design space was sampled was extremely low, calling into question the validity of the MI results. Table 9 details the MID results of the 97-D LTA HWB design space. When examining these results it is important to note that the Face Centered points had a much higher infeasible rate (around 30 percent) compared to approximately 10 percent for the PMC designs and thus it makes sense that they should point toward a more NHC space. Ultimately, when lumping both sets together, the MID value was reduced significantly (likely effect of the individual design variable distributions becoming more uniform as corner points were lumped with space-filling ones) yet still indicated a NHC classification of the

Table 8: Summary of Constraint Defined Feasible Set (CDFS) Classifications for the 97-D LTA HWB Design Space

	Training Set		Validation Set	
CDFS Name	Feasible	Infeasible	Feasible	Infeasible
Global	8576	1424	4610	390
FLOPS-ZFW	8576	1030	4610	269
ANOPP	8576	135	4610	62
CONDOR	8576	60	4610	19
Cum-Noise	8576	44	4610	25
ROC	8576	57	4610	2
MDP	8576	48	4610	9
Thrust-Conv	8576	28	4610	0
WATE	8576	15	4610	0
Main-Conv	8576	7	4610	4

space. It must still be pointed out however, that even with this result, the LPD for the sample is 1.1 (due to the high number of dimensions) lower than almost all critical LPD values identified in experiment 1 and thus making the NHC classification of the design space using MID tenuous at best. However, with this existing data set this is the information available, and given the percentage of failures/infeasible designs present, BAS has yet to be dismissed as potentially advantageous for this design space.

Table 9: Summary of Mutual Information Delta (MID) Data for Hypercubic Classification of the 97-D LTA HWB Design Space

DOE Description	LPD	MID Mean	MID SEM	Classification
3k (Face Centered)	1.086	2.394	0.287	NHC
7k (PMC)	1.096	2.025	0.212	NHC
10k (Combined)	1.010	0.908	0.114	NHC

Moving forward with SeBBAS, the CDFSs were constructed with the 'initial' training set and bound with RFs for NHC variable identification. It was here that the measurable cracks began to show for this DSE test in 97-D and only 10,000 training points. NHC variable isolation was extremely difficult leading to CDFS boundings constructed with over 30 variables each. Figures 68 - 71 illustrate a few of the variable importance rankings produced by the RF classifiers fit to the CDFSs in all of the design variables. While some variables are seen as extremely relevant like FCDSUB, TWR and FRFU, it is difficult do discern from visual inspection where to draw the cutoff in terms of how many variables should be included in the CDFS bounding classifiers. It is desired to construct these boundings with as many factors as necessary to capture the features which may exist in the design space, yet the with the inclusion of each additional variable, for the fixed training set, the resolution (LPD) suffers drastically. Here the curse of dimensionality again makes itself known and presents a difficult challenge to the construction of accurate CDFS bounding classifiers.

Although it was difficult to discern the most appropriate subset of variables to include in the CDFS boundings, The variable ranking results produced by the methodology are valuable in that they represent a knowledge gain concerning the design space currently unattainable through current DSE practices. Such information is very useful for determining causality within an experimental environment. These rankings can be used for each individual CDFS to backtrack through the experimental apparatus what settings may be the responsible for or driving particular failure modes. These results also serve as verification for whether or not appropriate physics-based relationships are responsible for boundaries within the design space. For the FLOPS-ZFW and ROC CDFSs, it is reassuring that propulsive and aerodynamic parameters are the factors seen mostly relevant for features within the design space, while more engine specific and noise governing parameters are most relevant to the ANOPP CDFS

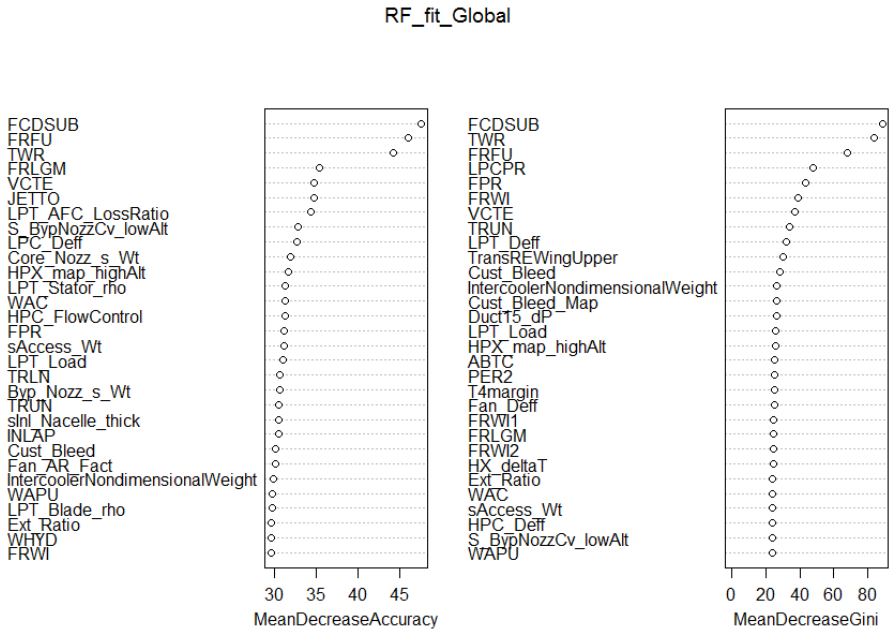

Figure 68: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding the 97-D Global Feasible Space

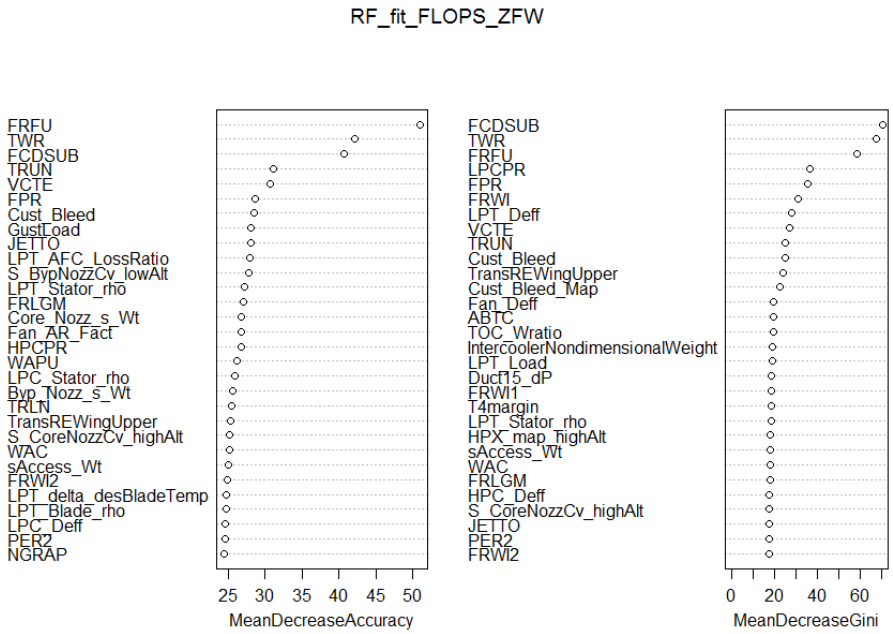

Figure 69: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding the 97-D FLOPS-ZFW CDFS

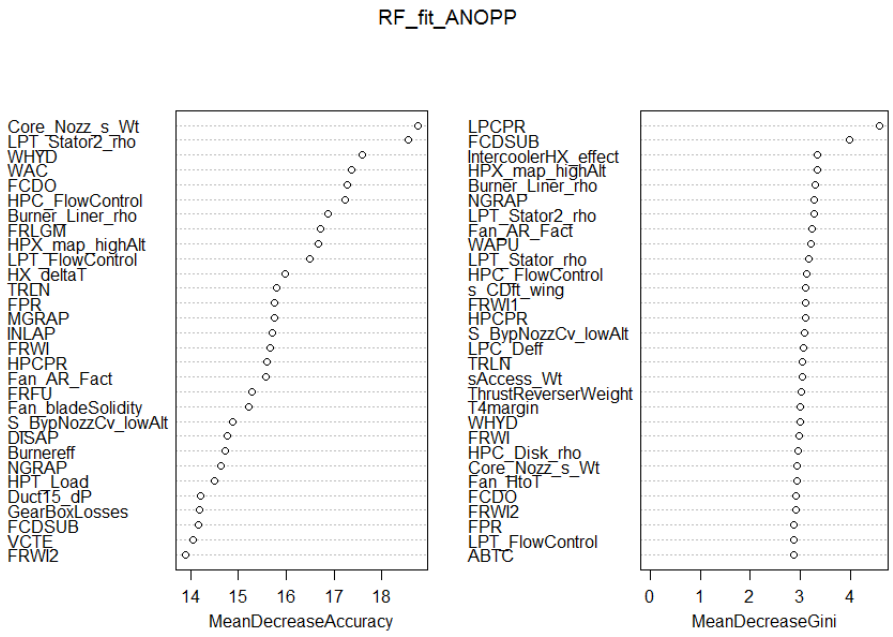

Figure 70: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding the 97-D ANOPP CDFS

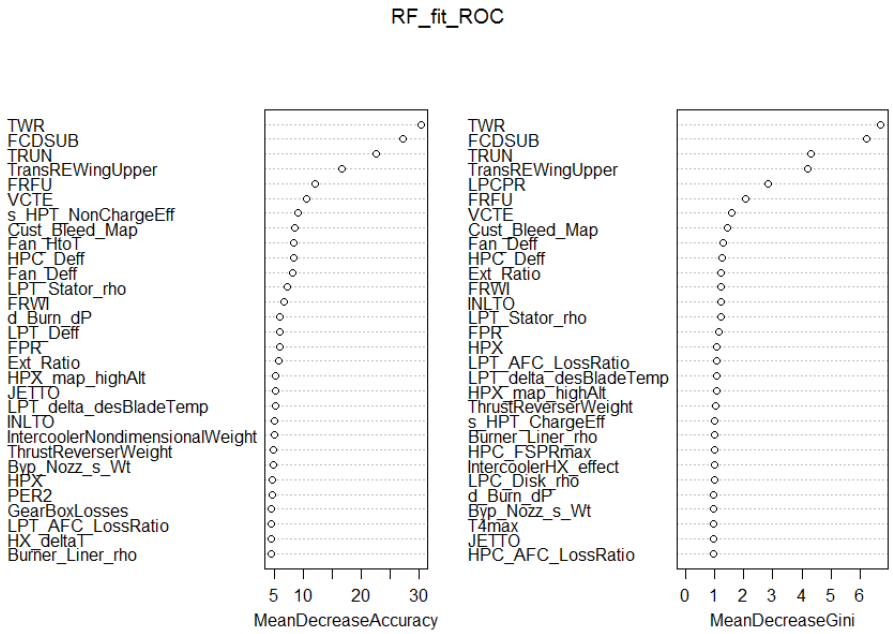

Figure 71: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding the 97-D ROC CDFS

boundaries. If an unexpected factor were identified this could indicate a numerical or otherwise non-physics-based issue (bug) present within an experimental apparatus.

Because the variable importance rankings did not provide definitive evidence of exactly which design variable subsets were most appropriate for the individual CDFS boundings, a sweep of factors was conducted for use in bounding the FLOPS-ZFW and ANOPP CDFSs. Figures 72 and 73 detail the performance of each of the SVM boundings conducted with differing subsets of design variables chosen. The cross-validation error (CVE - seen as squares within the figures) was tracked for each of these boundings and lower CVE generally reflected a more accurate bounding. The accuracy of each bounding classifier was tracked through the NFP and NFN generated for the validation sets utilizing the boundings to make predictions.

Results Interpretation (Fig. 72-73 and 78-79) The following performance figures illustrate how well BAS is able to reduce the number of misclassified designs for a bound NHC feasible design space as well as the corresponding cross validation error for a given bounding classifier. On the x-axis, the number of factors (or DV) used in constructing the specific bounding classifier is plotted. The factors to be included for any given bounding are ranked by increasing Gini impurity with the top subset selected as 'relevant' to construct a given bounding classifier. As the number of factors included increases, so does the complexity of the classifier (able to capture more effects), but because it is trained with only a fixed set of data, increasing the factors also effectively decreases the sample resolution for the design space. On the right y-axis, this effect can be seen in the behavior of the cross validation error. The best bounding classifiers, which minimize CVE, strike a balance between a parsimonious model and one which includes enough factors to observe the primary NHC features. On the left y-axis, the number of misclassified designs is tracked. To provide a reference for the number of misclassified designs when sampling BAU, a

solid horizontal line is plotted. All of these misclassifications are false positives as a Hypercubic sampling technique does not exclude any of the design space. Red dots on the chart represent the number of false negatives (designs thought infeasible when truly feasible) produced through BAS while blue dots represent the number of false positives (designs thought feasible when truly infeasible). The black dots are the sum of these two misclassification types. For BAS to be shown to be superior to sampling BAU, it must illustrate a lower total number of misclassified designs (i.e. be below the horizontal line).

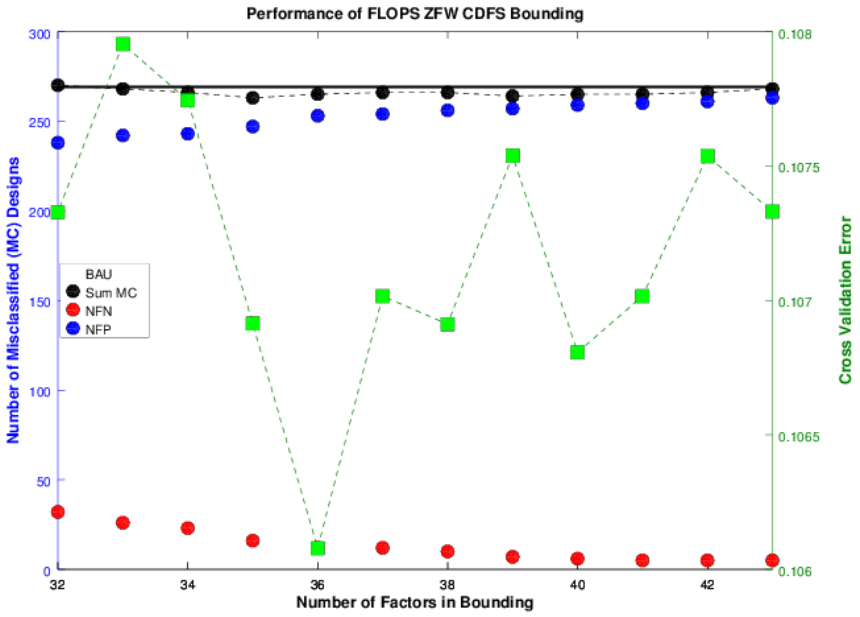

Figure 72: Performance of the SVM Classifiers Bounding the 97-D FLOPS-ZFW CDFS

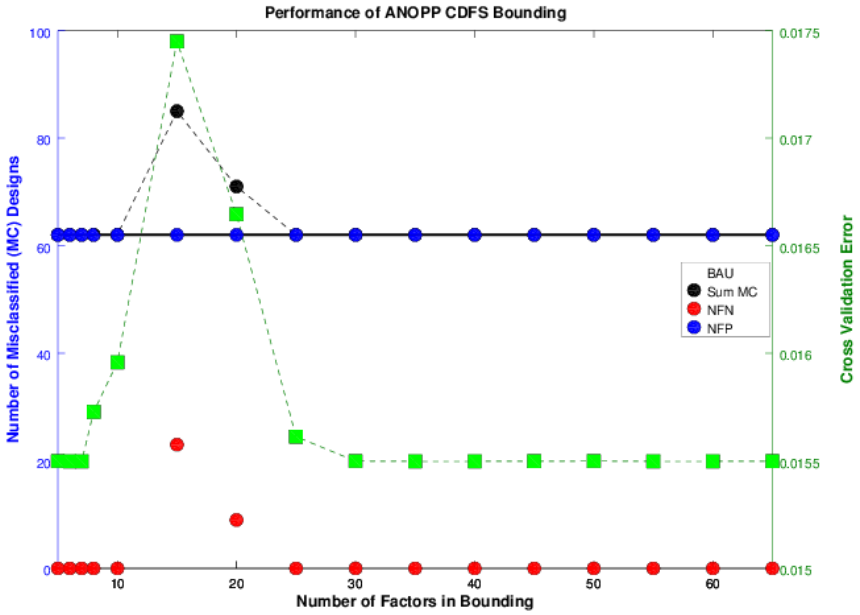

Figure 73: Performance of the SVM Classifiers Bounding the 97-D ANOPP CDFS

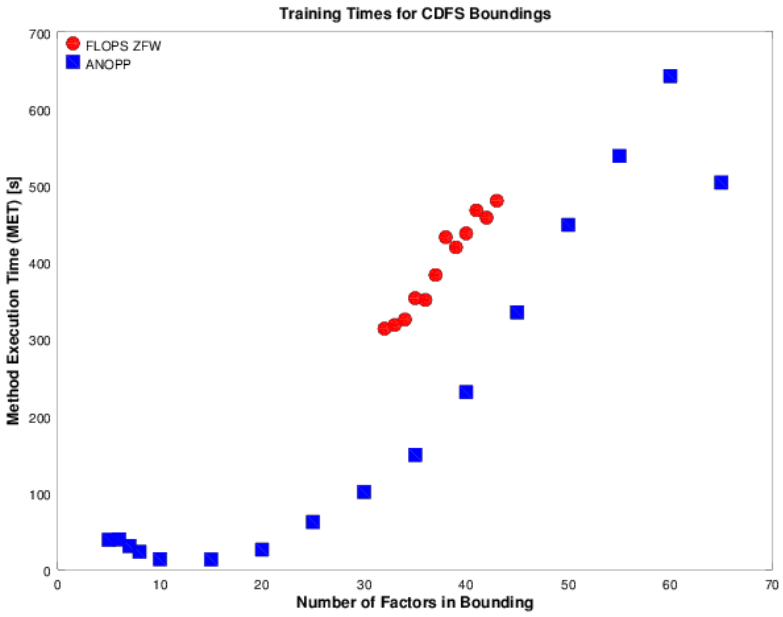

Figure 74: Method Execution Time (MET) Required for SVM Classifiers Bounding the 97-D CDFSs

Unfortunately even utilizing the most accurate bounding produced for the FLOPS-ZFW CDFS, the mislassifications illustrated that the infeasible set, compared to that achieved through PMC sampling, would only have been reduced by 22 design cases (only an 8 percent improvement) while 16 feasible cases would have been wrongfully excluded. The ANOPP CDFS boundings were unable to show any increase in sampling efficiency, and as a function of certain factor subsets even illustrated boundings that would yield worse results (through design space restriction) than sampling BAU. Computational resource expenditure tracked through MET and visible in Fig. 74 illustrated that while not certainly not prohibitive, the construction of the boundings, especially with large numbers of factors was not trivial, and thus could not be justified without demonstrated and significant improvement over BAU DSE practices. From these results it was clear that the methodology could not be used to appreciably show improvement over BAU DSE techniques for this particular design space. It was hypothesized that the low resolution used to sample the high-dimensional

space combined with the unbalanced nature of the classification results prevented the methodology from being effective. For this reason, the remaining steps in the methodology were suspended and adaptive samples were not created and evaluated for this design space. The hypothesis was considered unsubstantiated by these results, but not dis-proven. As a result, the 50-D DSE test was devised to offer a realistic yet perhaps more favorable design space with which the methodology would again attempt to illustrate benefit over common DSE practices.

5.4.4.2 Final Investigation: 50-D LTA HWB Design Space

The 50-D design space was created as a realistic DSE problem which also alleviated some of the factors which were believed to prohibit the methodology from illustrating benefit with the 97-D DSE test. Firstly, instead of post-processing an existing and non-homogeneously sampled design space, the new design space was explored in EDS utilizing PMC sampling for all DOEs. Additionally four separate DOE would be utilized to examine this space. Named Run1, Run2, ASE and ASR these DOE were used respectively to gather an initial training and validation set, improve resolution for MI and add additional validation data, quantify the increase in DSE efficiency through use of BAS, and ultimately adaptively refine the CDFS boundings created from the initial set.

Examining the classification results for Run1 (Table 10), an immediate difference can be recognized in terms of the balance of the data set. The entire data set output contained approximately 30 percent infeasible cases as opposed to approximately 12 percent observed in the 97-D DSE test. This increase in infeasible case percentage was likely driven by the inclusion of the 'TO Thrust' variable to this design space as well as the decreased lower limit on the TWR variable. As evidenced by all previous experiments this greater balance suggests that if NHC features are present they will likely be easier to detect using the methodology at a given resolution. Based upon lessons learned in the 97-D DSE test, another important realization was made through observation of this data in that some CDFSs (those with little infeasible classification data) would likely be unable to be resolved for this design space. Thus the decision was made to ignore all CDFSs except the two largest (FLOPS-ZFW and ROC) for the remainder of the experiment.

After the Run1 data set was classified, MI could be calculated for the purposes of Hypercubic Classification. The reduction in dimensions from 97 to 50 allowed the design space resolution to be increased significantly for the same number of design

Table 10: Summary of Constraint Defined Feasible Set (CDFS) Classifications for the 50-D LTA HWB Design Space

	Training Set		Validation Set	
CDFS Name	Feasible	Infeasible	Feasible	Infeasible
Global	7040	2781	3583	1330
FLOPS-ZFW	7040	2514	3583	1214
ROC	7040	246	3583	97
CONDOR	7040	199	3583	66
MDP	7040	11	3583	7
NPSS	7040	5	3583	8
Main-Conv	7040	5	3583	3
Thrust-Conv	7040	0	3583	1

cases evaluated (15,000). In addition, to drive the LPD even higher an additional 15,000 design cases from Run2 were also utilized to capture how MID values changed for the design space with increasing resolution. Table 11 displays the results of the MI Hypercubic classification on the 50-D design space. The first detail to note is that the resolution for the design space sample (measured in LPD) is now comfortably above some of the critical values required to resolve (using PMC sampling) some of the larger features potentially present within NHC design spaces (for example NLC2 and CBC2 which denied 20 and 50 percent of the feasible space respectively). Thus it can be reasoned that if features of similar magnitudes and confined to a small number of dimensions were present within the design space they could be observed by MI classification with the available resolution. Interestingly enough, not only do the large MID values calculated suggest the space is NHC, but when the resolution is increased, the MID values with error appear to increase, suggesting the space is likely NHC (based on trends observed in experiment 1). Therefore the 50-D design space was classified as NHC and passed to the next steps of the methodology.

Table 11: Summary of Mutual Information Delta (MID) Data for Hypercubic Classification of the 50-D LTA HWB Design Space

DOE Description	LPD	MID Mean	MID SEM	Classification
10k (PMC-Run1)	1.202	2.525	0.087	NHC
15k (PMC-Run2)	1.212	2.460	0.181	NHC
25k (PMC-Run1-2)	1.224	2.6385	0.188	NHC

With the 50-D LTA HWB design space classified as NHC and deemed a suitable candidate for BAS, CDFSs were constructed for both the FLOPS-ZRW and ROC constraint sets. With these two sets representing the large majority of the infeasible designs present within the overall set, their intersection was deemed an appropriate approximation of the global feasible design space. With this data binned as such, RFs were then fit to the CDFS for purposes of variable importance identification to the NHC CDFS. Figures 75 - 77 illustrate the variable relevance findings of these RF classifiers and tell a rather different story compared to the 97-D design space. For both the FLOPS-ZFW and ROC CDFSs, a small subset of variables can be identified as most important to bounding the features defined by the CDFSs. Of key and unsurprising importance is the TWR variable which had its lower limit reduced significantly for this design space. Also present in a more subtle way is the TO Thrust variable as well as other familiar variables associated with aerodynamics, weight or propulsive performance (ex: FCDSUB, FRFU and Cust-Bleed-Map).

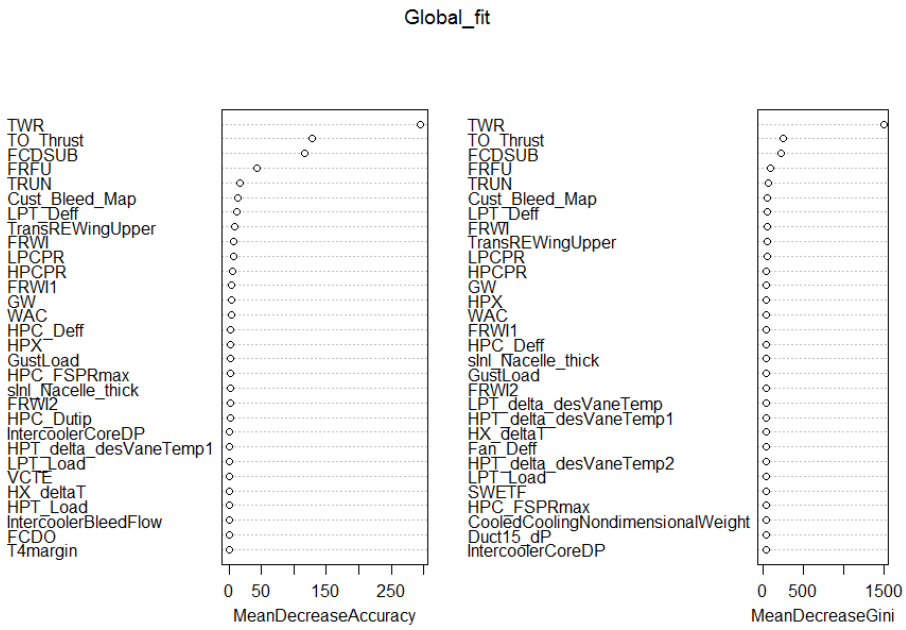

Figure 75: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding the 50-D Global Feasible Space

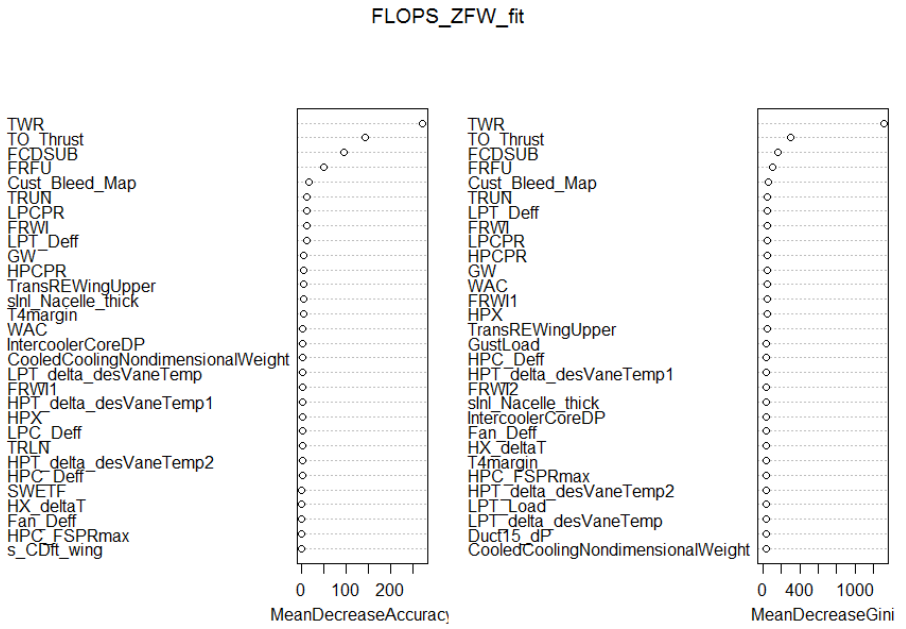

Figure 76: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding the 50-D FLOPS-ZFW CDFS

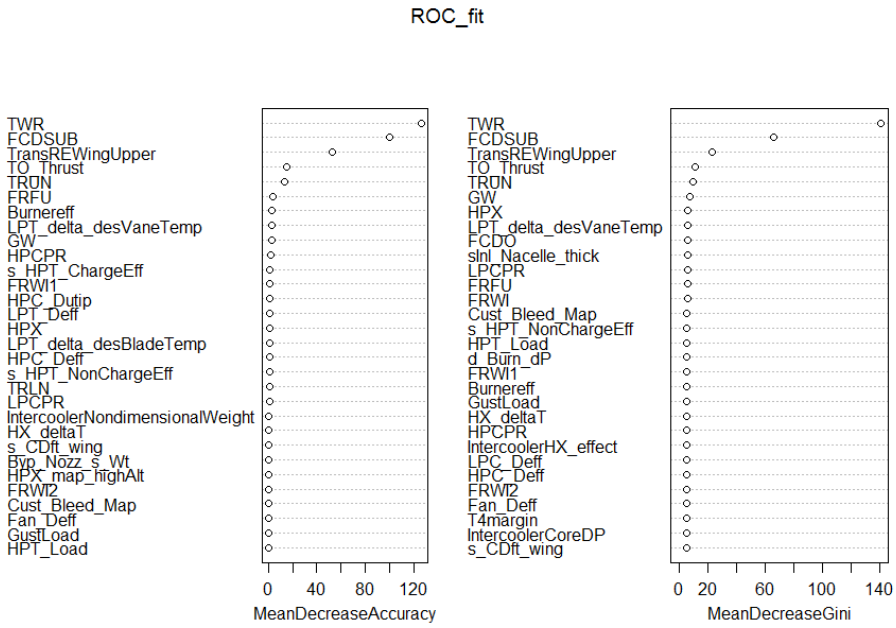

Figure 77: Variable Importance Rankings from RF Classifiers Used to Identify Variables Relevant to Bounding the 50-D ROC CDFS

Again utilizing lessons learned from the investigation of the 97-D LTA HWB design space, cross-validation error was minimized to identify the subset of factors/variables which would produce the most accurate CDFS boundings with the given design space resolution provided by the 'initial' training set of the Run1 DOE. Figures 78 and 79 illustrate the performance of the sweep of boundings investigated for the CDFS validated against the Run1 validation set. As with the results from the 97-D space, the lower the CVE, the more accurate the boundings were for representing the CDFSs. Unique to the 50-D DSE test however, the boundings showed significant improvement over sampling BAU. Compared to that achieved through PMC sampling, the optimal bounding for the FLOPS-ZFW CDFS illustrated a theoretical elimination of 945 false positive design cases (a 77.8 percent improvement) while 417 feasible cases would have been wrongfully excluded (when referenced against the Run1 validation set). The ROC CDFS boundings were also able to show an increase in sampling efficiency, yielding a theoretical elimination of 84 percent of false positives at the expense of classifying only 8 false negatives out of the entire ROC CDFS validation set of 3680 cases. These results showed that the boundings constructed for the CDFS, while not perfectly accurate, had the potential to improve over BAU sampling practices for a design space with these characteristics.

Furthermore as the optimal boundings required much less features than those found in the 97-D DES test, the computationl resources required to construct them were much less significant as evidenced by the MET values presented in fig. 80. Such low training times allow for sweeps to be performed to search for optimal feature selection settings without being prohibitive. It was hypothesized that the difference in training timese required between the FLOPS-ZFW and ROC CDFS boundings stems from the greater amount of training data and more balanced classification results present within the FLOPS-ZFW CDFS.

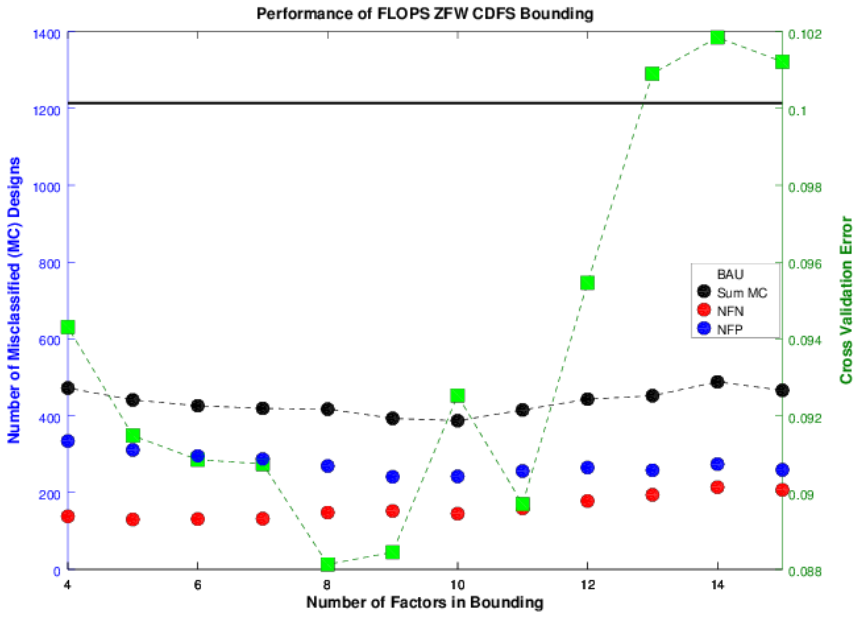

Figure 78: Performance of the SVM Classifiers Bounding the 50-D FLOPS-ZFW CDFS

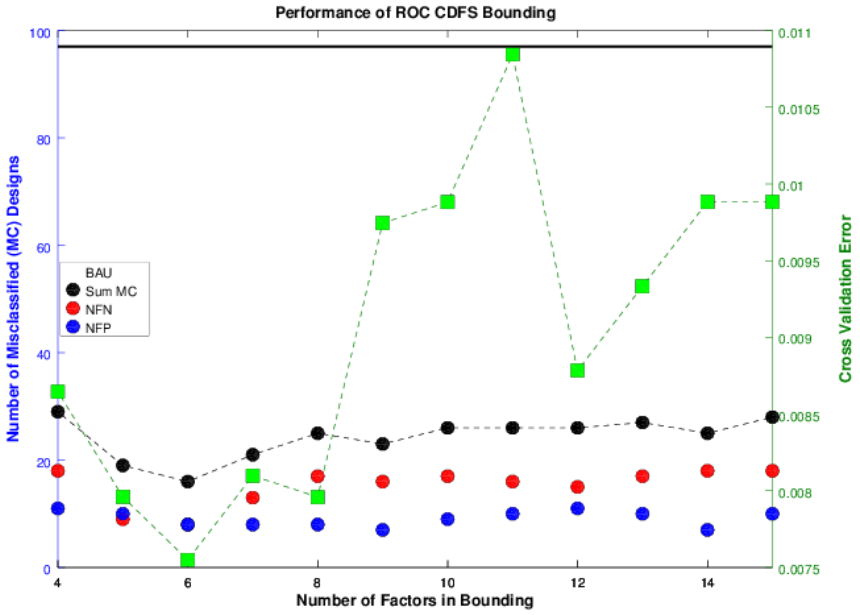

Figure 79: Performance of the SVM Classifiers Bounding the 50-D ROC CDFS

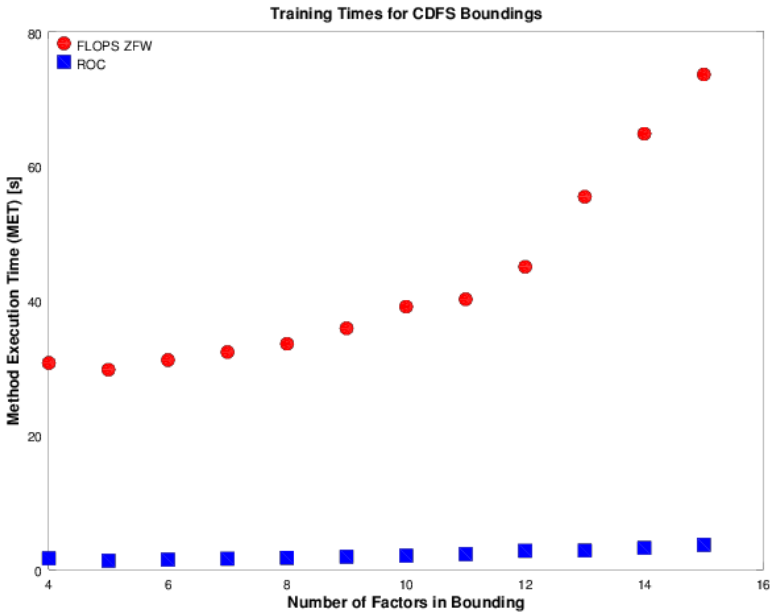

Figure 80: Method Execution Time (MET) Required for SVM Classifiers Bounding the 50-D CDFSs

With boundings for the CDFSs illustrating potential efficiency gains over BAU it was decided to continue with the final steps of the methodology which leveraged the boundings to generate adaptive samples for the NHC feasible space. Two separate DOEs were devised for adaptive sampling purposes. The first adaptive sampling DOE (ASE) was created to explore the global feasible space by querying the boundings of each CDFS simultaneously for designs which are classified as feasible by all boundings. To accomplish this, a dense PMC sample was generated for the design space and the new points were queried by each of the CDFS bounding classifiers to determine if the suggested design was likely feasible. Once 5000 designs were identified within the dense sample which appeared globally feasible, the ASE DOE was created from this set. The second adaptive sample DOE (ASR) was generated to refine the global NHC boundary by querying the boundings of each CDFS simultaneously for designs which were hardest to classify (classification uncertainty is high). This task was achieved by again leveraging the dense PMC sample for the design space and selecting new points which had classification probabilities for each grouping within some small threshold of each other. For example, if a point had a 0.45 classification probability as feasible and a corresponding probability of 0.55 as infeasible then the classifier could not determine with great confidence to which grouping the design likely belongs. Thus this design was assumed to be in the vicinity of the classifier boundary (or at least in a region of poor sample resolution) and would be likely to be selected for the ASR DOE. Again once 5000 designs were identified within the dense sample which appeared to be in the vicinity of the on of the CDFS classifier boundaries, the ASR DOE was created from this set.

Once the ASE and ASR DOEs were generated by leveraging the CDFS bounding classifiers, they were evaluated using EDS. Their outputs were then classified and utilized for their respective purposes. The classified output of the ASE DOE illustrated a decrease in global infeasible cases from approximately 30.0 percent in Run1

to 10.5 percent. This improvement in sampling efficiency was enabled by the combined CDFS boundings which caused the percentage of infeasible cases produced by the FLOPS-ZFW error to fall from 24.8 to 7.1 percent and those produced by the ROC error to fall from 2.2 to 0.2 percent. These results are summarized in Table 12.

Table 12: Summary of Reduction in Infeasible Design Cases Achieved through Bounded Adaptive Sampling (BAS)

	Run 1 Infeasible Cases		ASE Infeasible Cases	
CDFS Name	Number	Percentage	Number	Percentage
Global	4501	30.0	525	10.5
FLOPS-ZFW	3722	24.8	354	7.1
ROC	330	2.2	8	0.2

The classified output of the ASR DOE was used to refine the CDFS bounding classifiers initially generated in the 1st iteration of the method, as such it was added to the training data set from Run 1 and then utilized to train the 2nd iteration of CDFS bounding classifiers. Once these new classifiers had been trained they were compared against the 1st iteration utilizing the validation set from Run 2 to determine in the boundary had indeed been refined. Table 13 illustrates this comparison and the efficiency improvements achieved through BAS. Interestingly between the 1st and 2nd iterations the ability of the CDFS boundings to avoid suggesting infeasible designs is hardly changed however the bounding appears to become less conservative in nature as the NFN is decreased more significantly ultimately leading to a SBD bound global feasible design space subject to less mislassification.

With both knowledge gain observed through NHC variable identification as well as sampling efficiency improvement demonstrated through successful use of SBD BAS of CDFS, hypothesis 4 was considered substantiated. It is important to note that the 50-D DSE test was likely able to provide the methodology with an opportunity to demonstrate its merits due largely to the increased sample resolution (compared to

Table 13: Performance Comparison between CDFS Bounding Classifiers Generated in the 1st and 2nd Iterations of Bounded Adaptive Sampling (BAS) for the LTA HWB 50-D Design Space

BAS Iteration	NFP	NFN	Percent Feasible	Percent Misclassified
0 (Baseline PMC)	4075	0	72.0	28.0
1st Run1	899	464	93.8	9.4
2nd Run1+ASR	907	413	93.8	9.1

the 97-D space) and the more balanced classified data set which were both available. The last step performed in this experiment was simply to perform a visual check that the methodology had indeed done its job and thus a subset of the design space was visualized in 2-D and 3-D scatterplot matrices.

Results Interpretation (Fig. 81-88) These figures illustrate 2-D and 3-D projections of the 50-D design space in 11 and 3 variables of interest respectively (ranked highly in terms of relevance to the CDFSs). In these figures the green points represent feasible designs while the red points represent infeasible designs. It is first important to note that although much of the scatter of infeasible designs throughout the space may appear random, there may yet be structure in where these failed designs exist in the hypervolume which cannot be observed when the remaining 47 or 48 other dimensions are collapsed. Secondly, large concentrations of infeasibile designs do appear in regions of the design space observable in two or three dimensions. When BAS is performed, the adaptive samples which leverage the bounding classifiers show a much lower ratio of infeasible to feasible points and also avoid those regions where high densities of infeasible designs were encountered.

Looking at these figures it is immediately apparent how relevant TWR was to the CDFSs. However it is also interesting to note that there are many feasible designs with very low TWR values even though this region is prone to many infeasible designs. A simplistic but effective approach to increase the rate of return following BAS sampling

practices would be to increase the lower limit on the TWR variable, but doing so would drastically reduce the hypervolume able to be explored. Additionally, TWR is a design variable of great importance in aircraft conceptual design and one that is desired to be minimized for optimal aircraft sizing. Thus implementation of this methodology for DSE allows for exploration of potentially desirable regions of the design space with the ability to avoid infeasible regions that would otherwise make such large explorations prohibitively expensive. To illustrate the consequence of this ability, of the top ten designs throughout the design space in terms of minimizing block fuel burn, if the design space were truncated to the limits on TWR examined in the 97-D case (increasing the lower limit from 0.210 to 0.257) **half** of these top ten performing designs would be lost.

Another crucial takeaway from these visualizations is that some regions of the design space may be completely denied by the BAS approach as evidenced by the white voids appearing in the design space in the figures generated from the ASE DOE. While these denied regions may not be purely infeasible they do represent areas with large concentrations of infeasible designs and thus should be avoided by regression models. Often when regression or surrogate models are fit to a design space they are assumed valid over the entire Hypercubic design space, yet for a NHC design space that is certainly not the case. Another benefit of utilizing a methodology such as this one is that the CDFS boundings could be queried to ensure regression models fit to the data remain within the regions for which they can be assumed valid adding in a protection against model extrapolation.

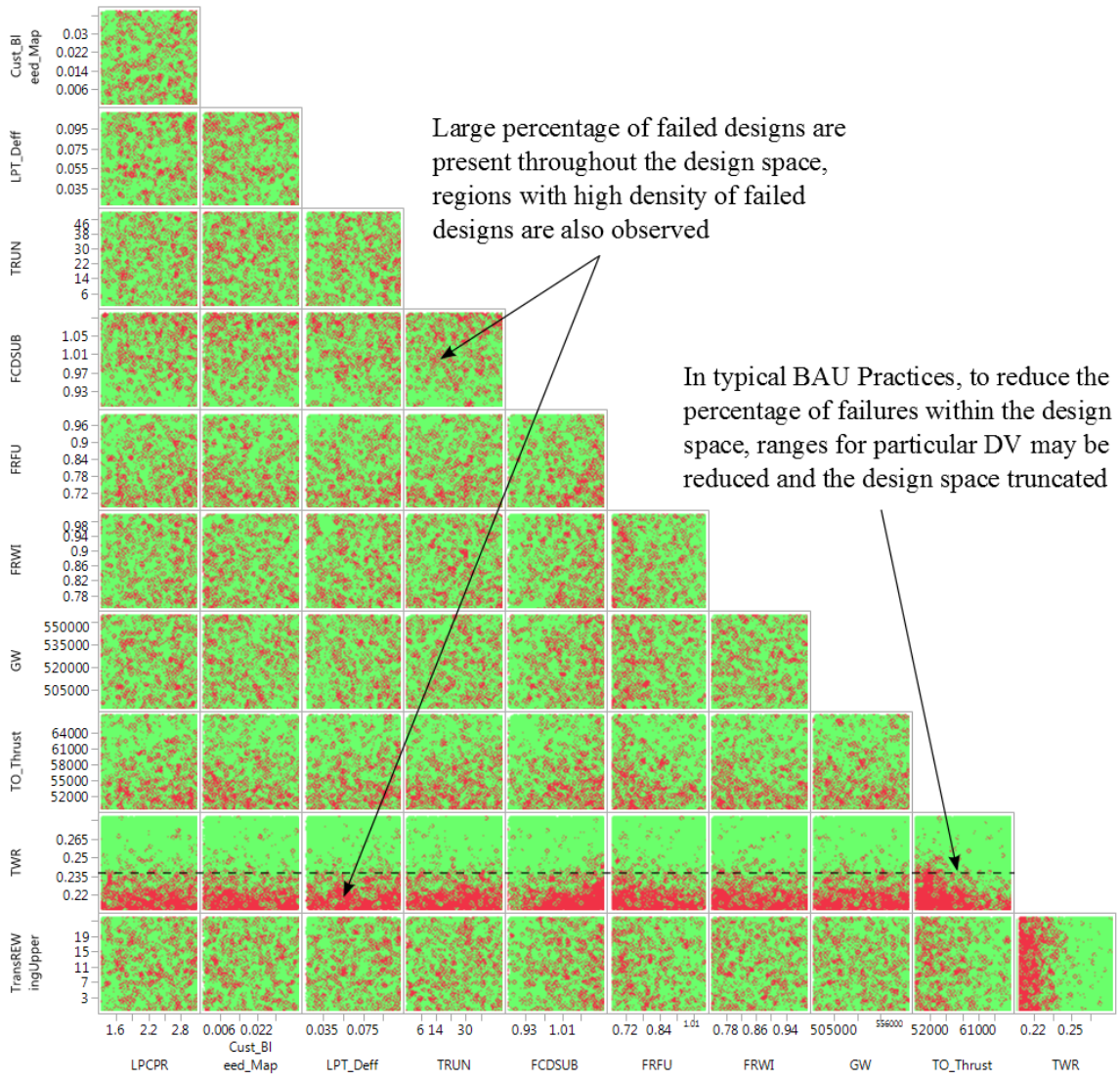

Figure 81: Scatterplot Matrix 2-D Design Space Visualization for the Global Feasible Space using Run1 Data (approx. 15000 cases)

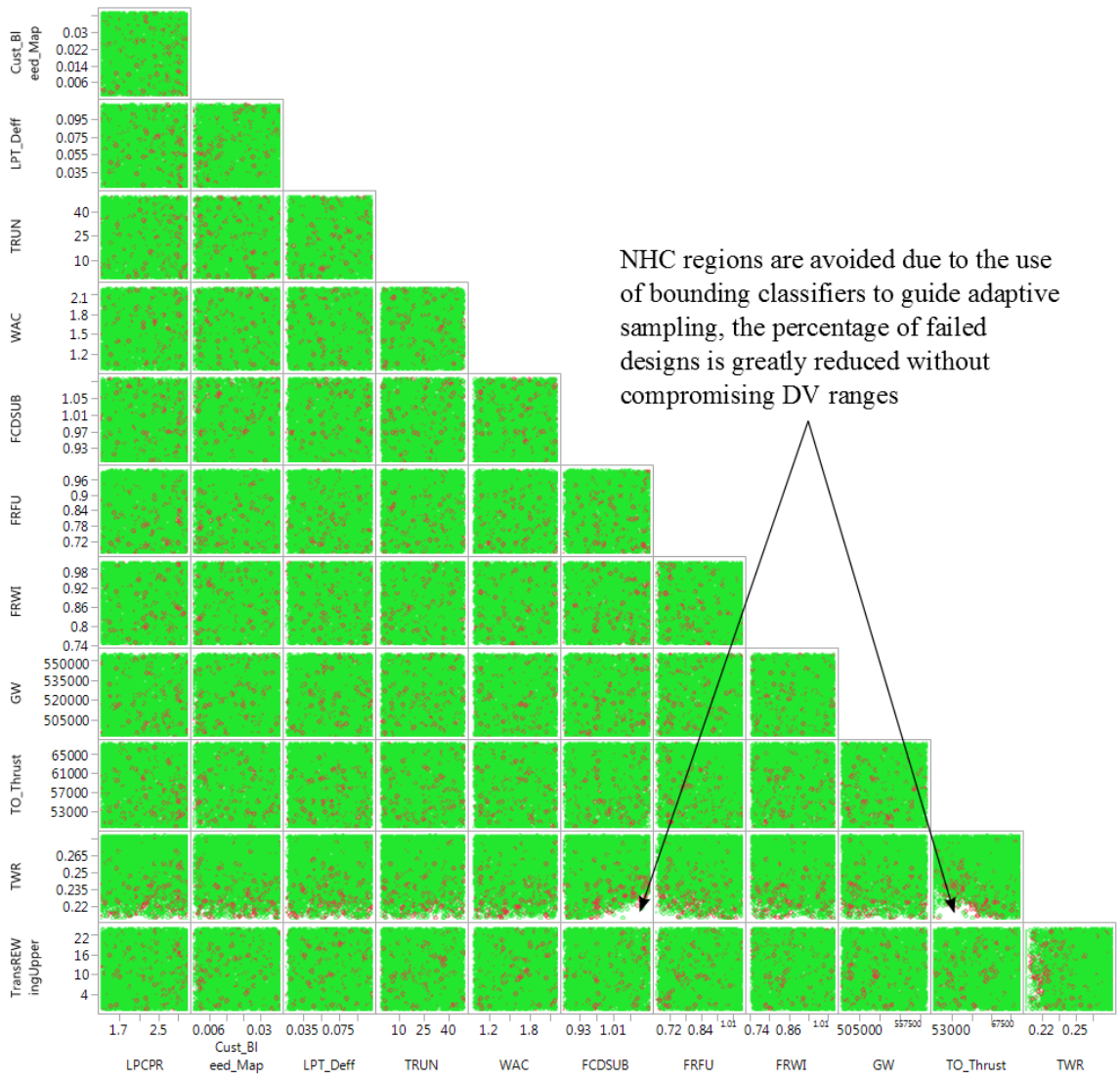

Figure 82: Scatterplot Matrix 2-D Design Space Visualization for the Global Feasible Space using ASE Data (approx. 5000 cases)

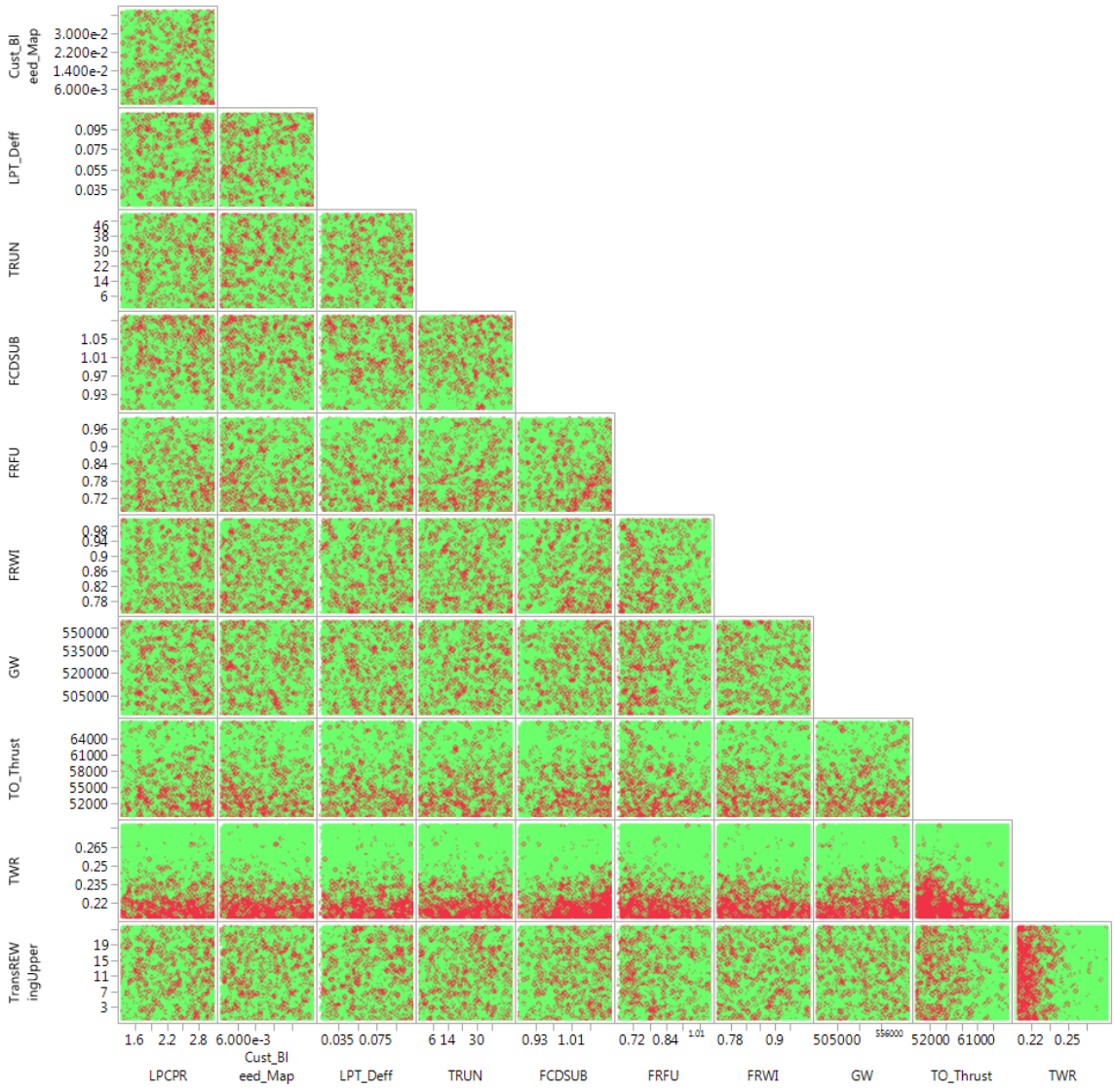

Figure 83: Scatterplot Matrix 2-D Design Space Visualization for the FLOPS-ZFW CDFS using Run1 Data (approx. 15000 cases)

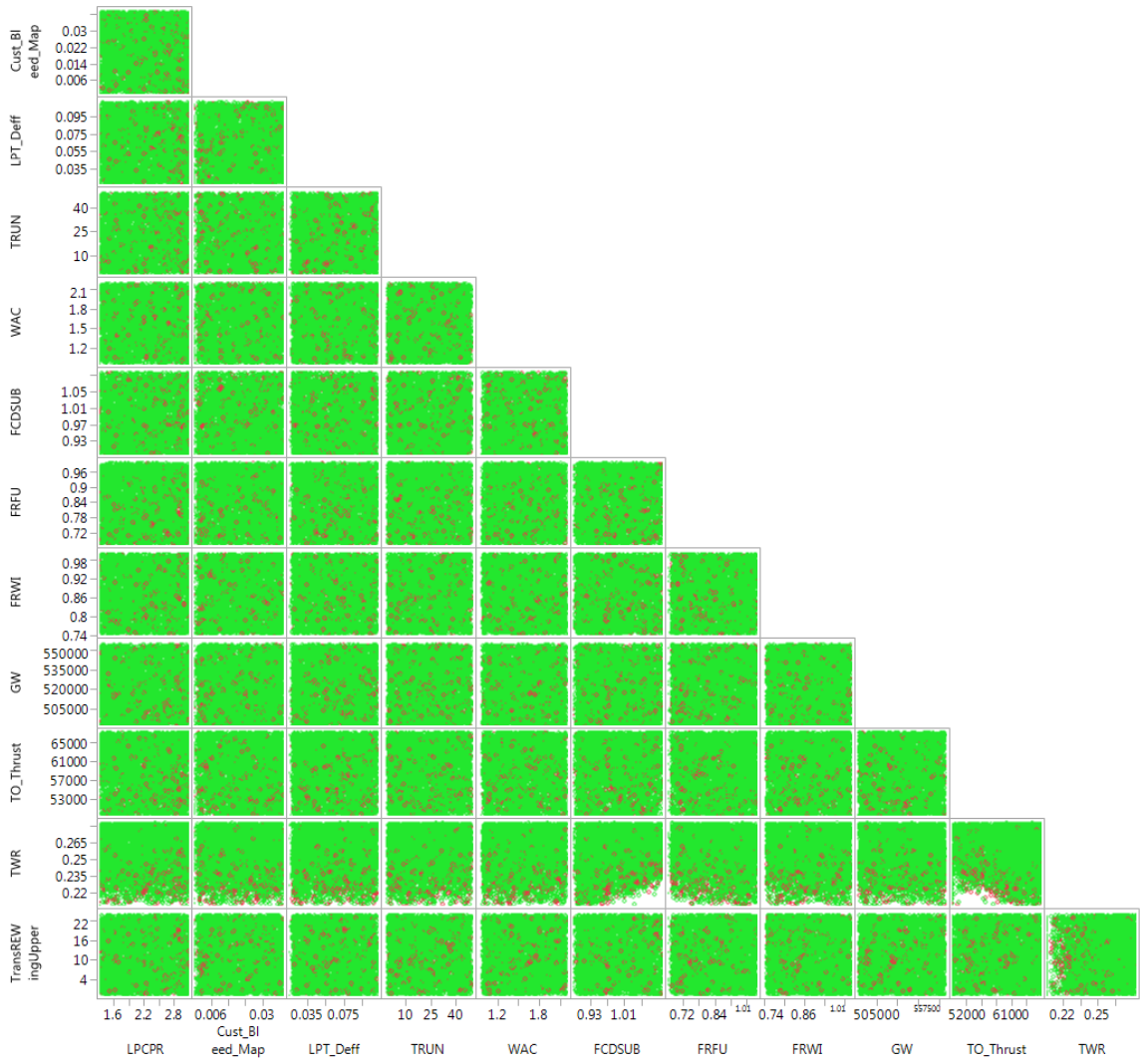

Figure 84: Scatterplot Matrix 2-D Design Space Visualization for the FLOPS-ZFW CDFS using ASE Data (approx. 5000 cases)

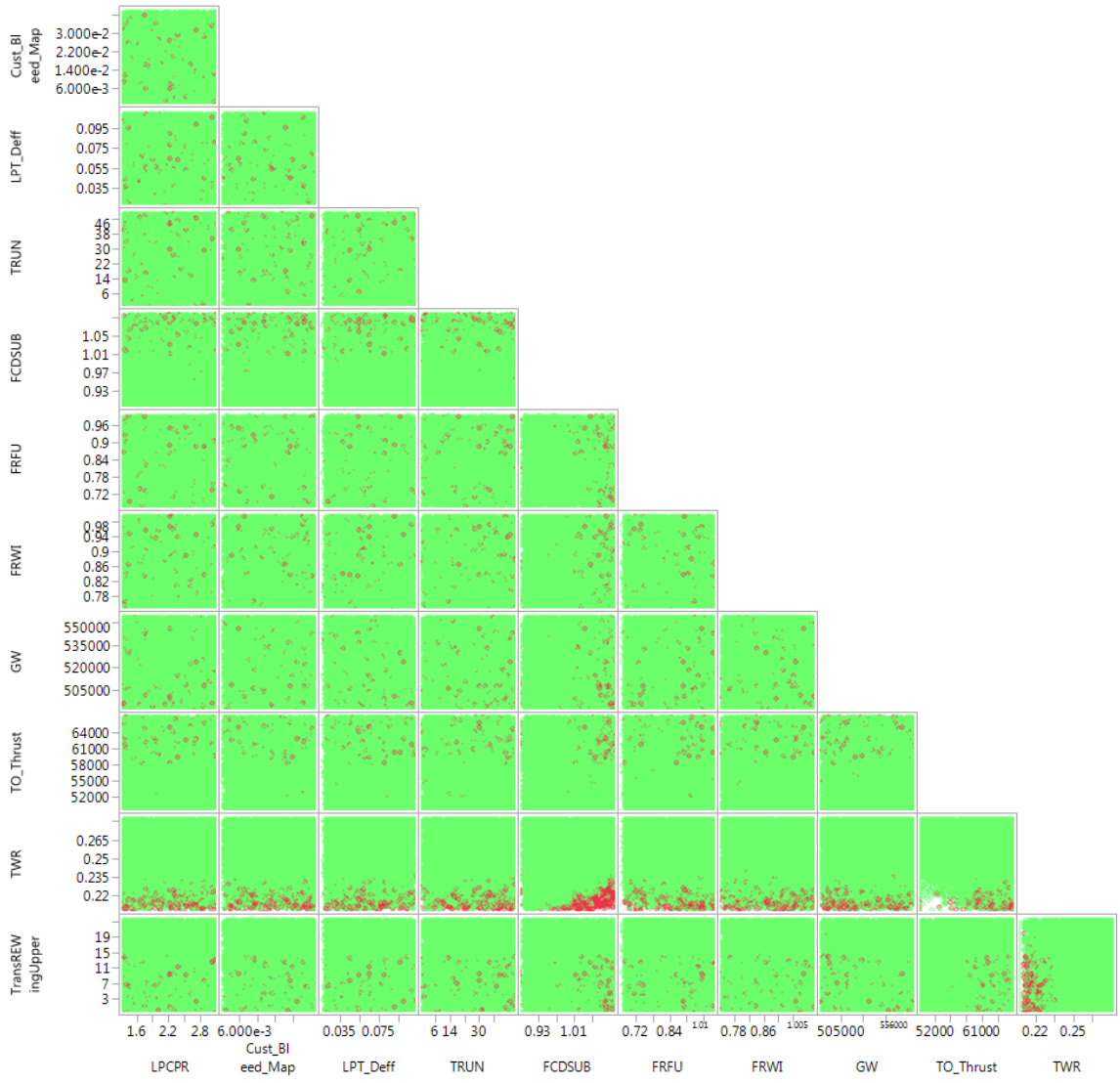

Figure 85: Scatterplot Matrix 2-D Design Space Visualization for the ROC CDFS using Run1 Data (approx. 15000 cases)

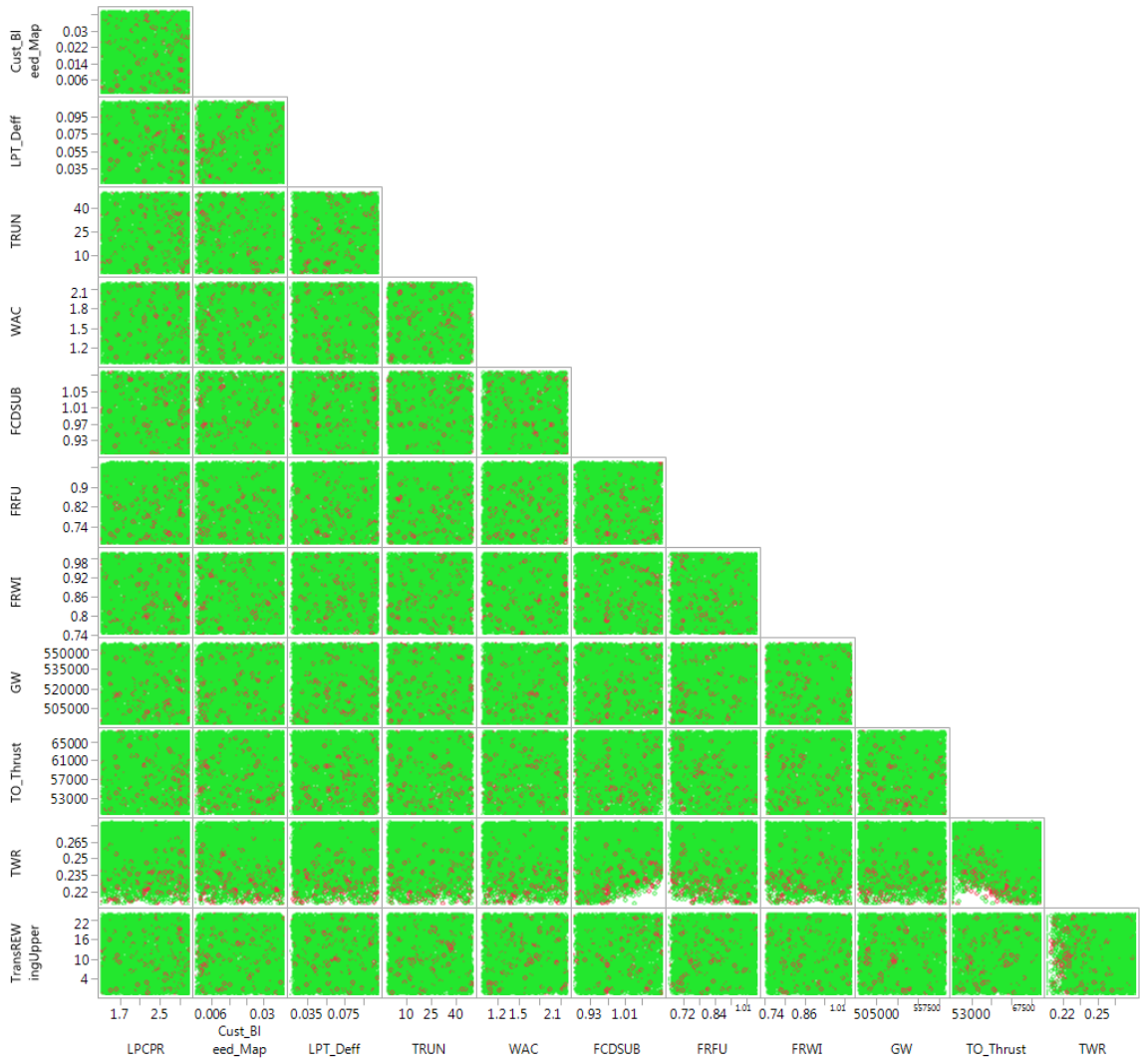

Figure 86: Scatterplot Matrix 2-D Design Space Visualization for the ROC CDFS using ASE Data (approx. 5000 cases)

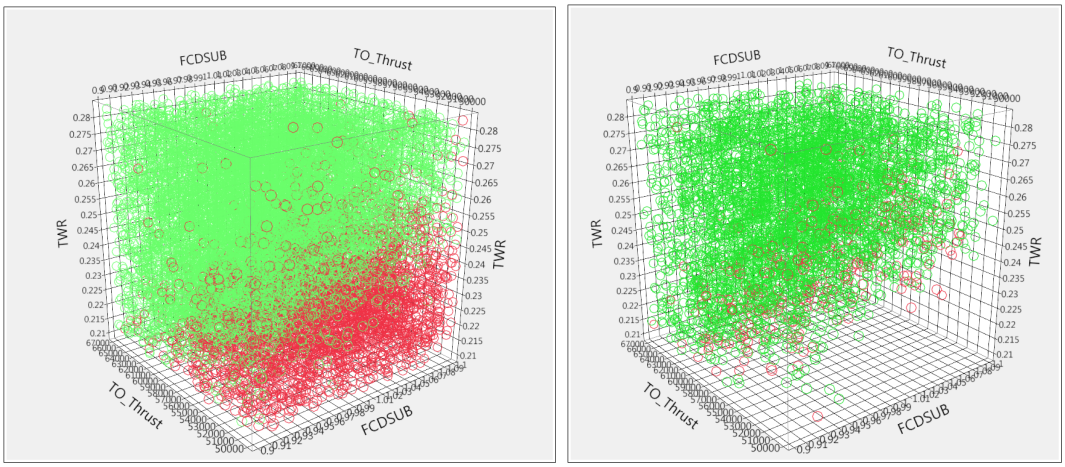

Figure 87: 3-D Design Space Visualization for the FLOPS-ZFW CDFS using Run1 (L) and ASE (R) Data

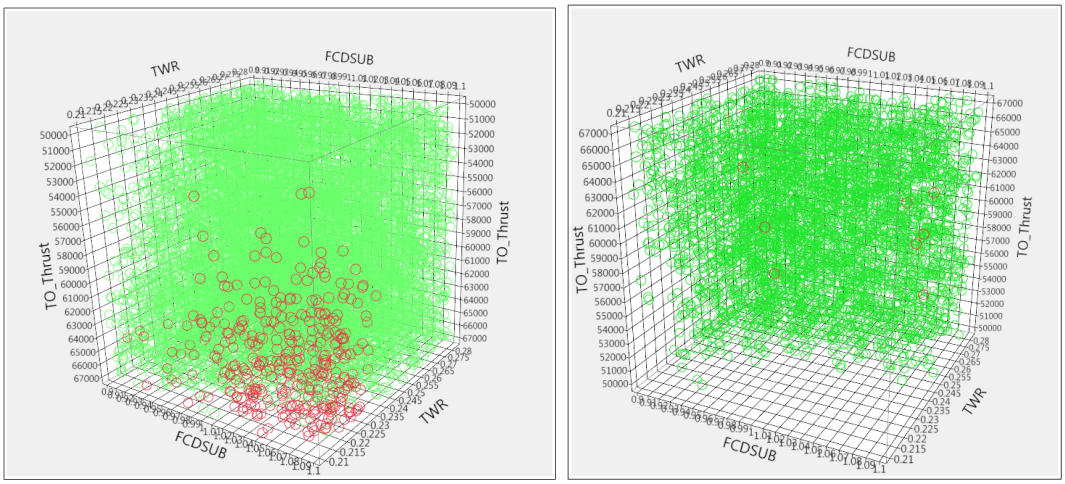

Figure 88: 3-D Design Space Visualization for the ROC CDFS using Run1 (L) and ASE (R) Data

5.4.5 Conclusions and Consequences

Results for experiment 4 were produced through the implementation of the thesis methodology on two different design spaces explored using EDS version 5.4 for the conceptual design of an LTA HWB aircraft concept. The two design spaces differed in number of dimensions (97-D and 50-D) and resolution with which they were sampled (LPD: 1.010 and 1.202 respectively). The methodology ultimately failed to produce any appreciable improvement compared to BAU DSE techniques like PMC sampling for the 97-D design space. This failure is attributed the low resolution with which this design space was sampled and the imbalance of the classified output which contained relatively low percentages of infeasible designs vital to training accurate bounding classifiers. The 50-D design space was created and explored using the methodology and learning from these failures. Ultimately his test was able to show the capacity for significant improvement over BAU sampling with the adoption of the SBD BAS approach within the methodology. Hypothesis 4 and the thesis methodology in general was considered substantiated based upon these results, but the 97-D DSE test also provided evidence for when and how the methodology could fail.

CHAPTER VI

CONCLUSION

6.1 Summary of Contributions

This thesis through experimentation and survey and synthesis of relevant literature has yielded the following contributions:

- Evidence of use of MI as a Classifier for Non-Hypercubic experimental design spaces (EXP 1)
- Establishment of effective Levels Per Dimension (LPD) as a similarity parameter for distinguishing features in the design space (EXP 1)
- Informed DSE Guidance as a function of design space characteristics and relative expense/consequence (synthesis of literature and own ideas)
- Illustration of the advantages of using a set-based as opposed to a global approach for determining the boundary of NHC feasible design spaces (EXP 3)
- Bounded 50-D LTA HWB design space in EDS (illustrated knowledge gain, and failed case reduction on a real problem) (EXP 4)
- Set-Based Bounding and Adaptive Sampling (SeBBAS) Methodology (EXP 4, synthesis of literature and own ideas)
- The Design Space Exploration Decision Support Methodology (DSE-DSM) (Overarching Thesis)

6.1.1 Methodology Development

This thesis work produced a methodology which provides decision support for design space exploration efforts for arbitrary design spaces. This methodology illustrates a method to perform hypercubic classification as well as bounded adaptive sampling for design spaces for problems which are computationally expensive, non-hypercubic and subject to revisitation. It is important to note that this methodology is by no means restricted to the conceptual aircraft design problem and is applicable for all complex design problems with these attributes. Put simply, the main goal of the methodology is to drastically improve the success rate for design cases evaluated within the design spaces defined by such problems. Because these problems are expensive and yet many successful cases are still necessary to generate accurate surrogates or provide visualization, it is imperative that the limits of the feasible design space be known and understood so that computational resources are not repeatedly wasted exploring infeasible regions. The methodology presented herein provides a structured and somewhat robust means of accomplishing this goal.

6.1.2 Concluding Remarks

This thesis proposed a new methodology (DSE-DSM with SeBBAS) meant to provide design space exploration decision support and effective and efficient means of examining problems which are computationally expensive, non-hypercubic in nature and require revisitation. Motivating the development, testing and ultimate proof of concept of this methodology is a design problem characterized by these elements and concerned with the physics-based design of an advanced civil transport aircraft concept. This design problem ultimately seeks to estimate the performance benefits of such a concept and provide an assessment of its ability to help address some of the major issues facing civil aviation today, most notably, the desired reduction in aircraft fuel burn. Because of the aforementioned characteristics present within this design

problem, current methods involving the use of traditional DOE to sample the space and then generate surrogate models were found lacking in their ability to provide an efficient solution. For this reason, once a design space has been classified as non-hypercubic using a Mutual Information based test, the proposed methodology takes an adaptive approach in which a sophisticated set of boundings are constructed utilizing Machine Learning classifiers (random forests and kernel-based support vector machines) for the design space. These boundings are integrated to form an estimate for the global feasible design space which is then iteratively refined and exploited to improve the useful rate of return for a given experimental budget. Drawing from Set-Based Design techniques, this methodology uses the construction of constraint defined feasible sets $CDFS$ which are bound by classifiers and then 'intergrated by intersection' to discover the feasible design space. Leveraging this set-based approach and identifying the relevant variable subsets for the CDFSs allows for a superior representation of the global feasible space for a given experimental budget. This feasible space is what remains of the initially sampled hypercubic design space once all the constraints, effects of correlated design variables and regions of computational method infeasbility have been considered.

It was hypothesized that the representation of the design space and its defining characteristics produced by the methodology would be superior to that which could be obtained by contemporary methods for the same computational resource budget. This claim was substantiated through the testing of four hypotheses within four separate experiments aimed at different aspects of the problem considered and the proposed methodology.

Through Experiment I, Mutual Information MI was demonstrated to provide a useful means of hypercubic classification with the caveat that sufficient design space sampling resolution was used. To generalize this concept of sample resolution, a new similarity parameter was developed and coined effective 'Levels Per Dimension'

LPD. This parameter combines the effect of sample size and dimensionality into a single metric determining sampling resolution while accounting for the exponential growth of hypervolumes with the inclusion of additional dimensions. Using this metric and a binned MI estimator, critical resolution thresholds were quantified for various hypercubic and non-hypercubic constrained design spaces sampled by multiple initial DOE. It was shown that when utilizing a Pseudo-Monte Carlo DOE for sampling, an initial LPD of approximately 2 was sufficient for correctly resolving and classifying most features within a design space.

Experiment II illustrated the general benefit obtained through the use of a bounding classifier for the understanding and exploration of non-hypercubic design spaces. Again the hypothesis presented within this experiment was only considered substantiated when sufficient sample resolution was provided to the bounding classifier to resolve features within the space. Interestingly, these critical resolutions were found to be approximately those revealed in experiment I. Additionally, experiment II revealed that a higher dimensional space, with only a subset of its dimensions being non-hypercubic, can be seen as effectively collapsed from the point of view of a bounding classifier for the non-hypercubic regions. This revelation pointed to the potential for focused classifiers (like those constructed through set-based techniques) trained on only a subset of the design variables to effectively utilize a higher resolution than available to global classifiers with the same sample set.

A Set-Based approach was contrasted against a global approach for classifying and bounding the non-hypercubic feasible design space in Experiment III. This experiment illustrated that if the feasible regions formed by the constraint defined feasible sets are disjoint, then from the same set of training data, a smaller subset of variables can be identified that are relevant to the given CDFS and thus the other dimensions can effectively be collapsed. This collapsing of dimensions artificially increases the resolution (LPD) used to generate the bounding for that particular CDFS through

the reduction of the number of variables or features used to construct the bounding classifier. This results in set-based boundings that are more accurate (produce less false positives and false negatives) than their global counterparts while requiring not significantly more total computational effort to construct.

Ultimately, through the fourth and final experiment, the utility of DSE-DSM and associated SeBBAS approach were demonstrated for the conceptual design of a Large Twin Aisle LTA Hybrid Wing Body HWB aircraft within the Environmental Design Space EDS modeling and simulation environment. Given a design problem in 50 dimensions, the DSE-DSM methodology was able to increase the percentage of feasible designs achieved through designs space exploration from 72.0 to 93.8 percent when compared to Business As Usual BAU Pseudo-Monte Carlo PMC sampling after only a single iteration. Additionally, the methodology was able to identify and rank variables relevant to the non-hypercubic features present within the design space all without significant additional computational expense compared to BAU.

This thesis through the use of DSE-DSM and SeBBAS demonstrated the capacity for a more timely and resource conservative approach for the conceptual design of advanced aircraft concepts as well as other problems with similar characteristics. A capability to provide decision support for the exploration of arbitrary design spaces was presented along with a means to classify, bound and adaptively sample design spaces which require more advanced sampling techniques than provided by contemporary DOE methods. This methodology and the design space representation it provides allow for efficient surrogate generation, optimization, future design space exploration and visualization for problems which are computationally expensive, non-hypercubic and must be revisited.

6.2 Consequences

The functionality of the methodology and its representative elements was demonstrated through the experiments performed within this work. However, to convince a user to adopt such a approach and abandon BAU practices requires an acknowledgement of the consequences associated with its use. The following details some of the major consequences associated with the use of DSE-DSM and SeBBAS to guide DSE for generic design spaces:

- **More Efficient Resource Use**: One of the primary drivers for the use of this methodology was the promise of more efficient use of computational resources when performing DSE for NHC feasible design spaces. While computational expense is subjective, use of the methodology and in particular the SeBBAS method allows for significant potential savings of computational resources (quantified infeasible case reduction from 30.0 percent to 10.5 percent within the practical conceptual design problem). For design spaces with large NHC features, the methodology may likely enable DSE where previously infeasible due to unacceptably high percentages of infeasible designs.

- **Identification of NHC Variables**: Through the use of the Gini impurity and feature selection through cross-validation, the methodology was able to identify sets of design variables relevant to specific CDFSs. Because of this, for a given sample budget, the case density can be effectively increased through the collapse of dimensions irrelevant to the construction a bounding classifier for a particular CDFS. This ultimately allows for an improved understanding of a high dimensional NHC feasible design space to be attained with significantly less resources than would be possible when using a global approach. Furthermore, the extraction of these NHC variable sets can aid in understanding and debugging large multidisciplinary Modeling and Simulation environments.

- **Expansion of Design Variable Ranges**: The ability to adaptively sample the design space leveraging boundings allows for the expansion of the ranges on design variables. As the feasible/infeasible boundaries can be obtained, the ranges on each of the design variables can be expanded until these limits are encountered. This capability allows for a significant increase in the experimental hypervolume yielding the potential to explore much more of the design space and uncover desirable designs which were previously unattainable without expending significantly more computational effort.

- **Surrogate Monitoring for Extrapolation**: When surrogates/regressions are fit over a hypercubic space using business as usual practices, they are typically assumed valid over the entire hypervolume. However, if NHC regions exist where the design space is infeasible and/or no design points exist, then these regressions are extrapolating in these areas of the design space. With the use of the bounding classifiers provided by the methodology for NHC feasible spaces, surrogates/regressions fit to the hypercubic design space can be monitored to determine where they may be extrapolating and prevent incorrect conclusions from being drawn in these regions.

- **Bounded Optimization**: The bounding classifiers produced for NHC feasible spaces through use of the methodology effectively estimate boundaries for each CDFS. While the exact boundaries are not known with perfect confidence to the classifiers, the classifiers can be queried to determine if a given design likely belongs to a given CDFS. This information can be utilized to perform bounded optimization, allowing for infeasible/failed regions of the design space to be avoided while optimal designs are sought. This allows for optimization to be performed more efficiently, further conserving computational resources.

6.3 Future Work

Although all the hypotheses tested in this work were ultimately considered substantiated (with conditions) this by no means there is no more work left to do. The following details just a few of the potential areas for expansion and refinement of the ideas presented within this work:

1. **Surrogate Generation and Monitoring**: Generate surrogate models using the same computational budget with a traditional sampling approach and a DSE-DSM enabled approach. Compare surrogates for accuracy, are the DSE-DSM of better quality? Can the bounding classifiers for the CDFSs correctly show when surrogates are extrapolating?

2. **Imposition of Performance Constraints**: Impose performance constraints on the design space and evaluate the ability of SeBBAS to bound regions of the design space with high performing designs.

3. **Tune Classifiers**: Can the quality of the bounding classifiers produced by the RF and SVM methods be improved significantly through the modification of their respective tuning parameters and/or kernels?

4. **Expand Design Problem Ranges**: Can more NHC features/dimensions be identified within the 50-D design space examined for the LTA HWB if the design ranges are expanded?

5. **Design Space Analysis**: Determine the likely causes of the NHC regions discovered in the 50-D LTA HWB design problem with EDS. Are they due to physics, numerical issues or bugs within the code?

APPENDIX A

TEST PROBLEMS

The following contains graphical depictions and the MATLAB code used to generate all the test problems utilized within this work.

**Note for all code posted the following code applies:

```
n_val = 5000; % Number of design cases (5000 for display)
n_DV_val = 2; % Number of design variables (2 or 3 for display)
DOE_base = unifrnd(-0.5,0.5,n_val,n_DV_val); % Baseline PMC DOE
```

A.1 Hypercubic Constrained Design Spaces

The following test problems were representative Hypercubic design spaces featuring cases missing. These were used to test the MI classifier to determine if it could detect a hypercubic space (and not be fooled into thinking the space was non-hypercubic) even if some fraction of the baseline hypervolume was missing. In practical design problems, these spaces may be observed if design variables are examined over ranges in which they are infeasible. Additionally, if computational errors/instabilities cause truly random failures throughout the design space, then feasible spaces which remain hypercubic yet contain infeasible designs may also result.

A.1.1 Reduced Hypercube Single (RHS and RHS1)

This constrained design space featured a Reduced singular hypercube in all dimensions (RHS) and only the 1st dimension (RHS). This test design space was utilized to emulate design variables being evaluated over ranges for which they were infeasible. It is important to note that the infeasible region for a particular design variable was

purely independent of any other design variables. Therefore the feasible design space is reduced from the baseline hypercube, yet remains hypercubic in nature.

A.1.1.1 MATLAB code

```
% Reduced Hypercube Single (RHS)
rmv_Vol = 0.2;
Tgt_Vol = 1 - rmv_Vol;
HC_base = (Tgt_Vol)^(1/n_DV_val);
bound = HC_base/2;
DOE_RHS = [];
n_RHS = 0;
for i = 1:n_val
    if max(DOE_base(i,:)) <= bound && ...
            min(DOE_base(i,:)) >= (-1*bound)
        DOE_RHS = [DOE_RHS; DOE_base(i,:)];
        n_RHS = n_RHS+1;
    end
end

% Reduced Hypercube Single 1-D (RHS1)
rmv_Vol = 0.2;
Tgt_Vol = 1 - rmv_Vol;
HC_base = Tgt_Vol;
bound = HC_base/2;
DOE_RHS1 = [];
n_RHS1 = 0;
for i = 1:n_val
    if max(DOE_base(i,1)) <= bound && ...
            min(DOE_base(i,1)) >= (-1*bound)
        DOE_RHS1 = [DOE_RHS1; DOE_base(i,:)];
        n_RHS1 = n_RHS1+1;
```

```
    end
end
```

A.1.1.2 Graphical Representation

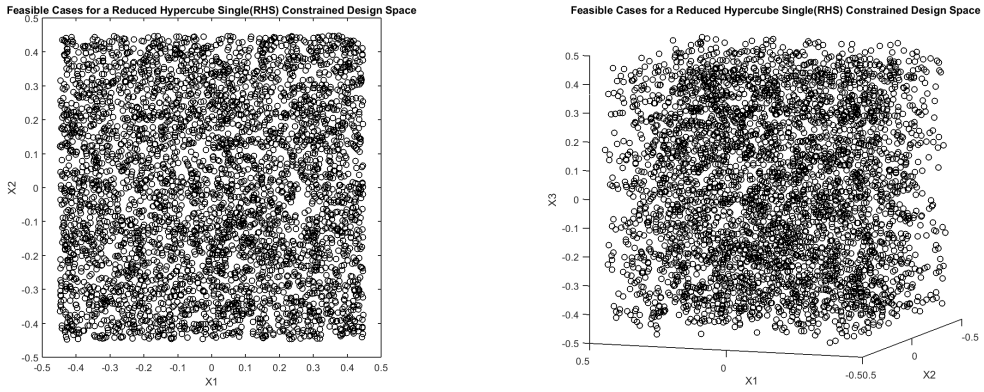

Figure 89: 2-D and 3-D Design Space Visualization for RHS Constrained Design Space

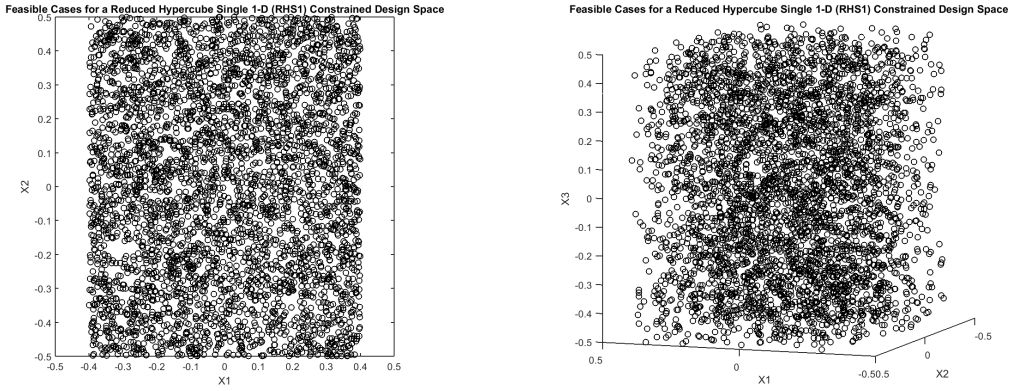

Figure 90: 2-D and 3-D Design Space Visualization for RHS1 Constrained Design Space

A.1.2 Reduced Hypercube Multiple (RMS and RMS1)

This constrained design space featured multiple Reduced hypercubes in all dimensions (RHM) and only the 1st dimension (RHM1). This test design space was utilized to emulate design variables being evaluated over ranges for which they were infeasible

resulting in multiple non-continuous feasible regions. It is important to note that the infeasible region for a particular design variable was purely independent of any other design variables. Therefore the feasible design space is reduced from the baseline hypercube, yet remains hypercubic in nature.

A.1.2.1 MATLAB code

```
% Reduced Hypercube Multiple (RHM)
rmv_Vol = 0.2;
Tgt_Vol = 1 - rmv_Vol;
HC_base = (Tgt_Vol)^(1/n_DV_val)/2;
min_b = -0.5 + HC_base;
max_b = 0.5 - HC_base;
DOE_RHM = [];
n_RHM = 0;
for i = 1:n_val
    keep_flag = 1;
    for j = 1:n_DV_val
        if DOE_base(i,j) > min_b && ...
                DOE_base(i,j) < max_b
            keep_flag = 0;
            break;
        end
    end
    if keep_flag == 1
        DOE_RHM = [DOE_RHM; DOE_base(i,:)];
        n_RHM = n_RHM+1;
    end
end

% Reduced Hypercube Multiple 1-D (RHM1)
```

```
rmv_Vol = 0.2;
Tgt_Vol = 1 - rmv_Vol;
HC_base = (Tgt_Vol)/2;
min_b = -0.5 + HC_base;
max_b = 0.5 - HC_base;
DOE_RHM1 = [];
n_RHM1 = 0;
for i = 1:n_val
    if DOE_base(i,1) > min_b && ...
            DOE_base(i,1) < max_b
    else
        DOE_RHM1 = [DOE_RHM1; DOE_base(i,:)];
        n_RHM1 = n_RHM1+1;
    end
end
```

A.1.2.2 Graphical Representation

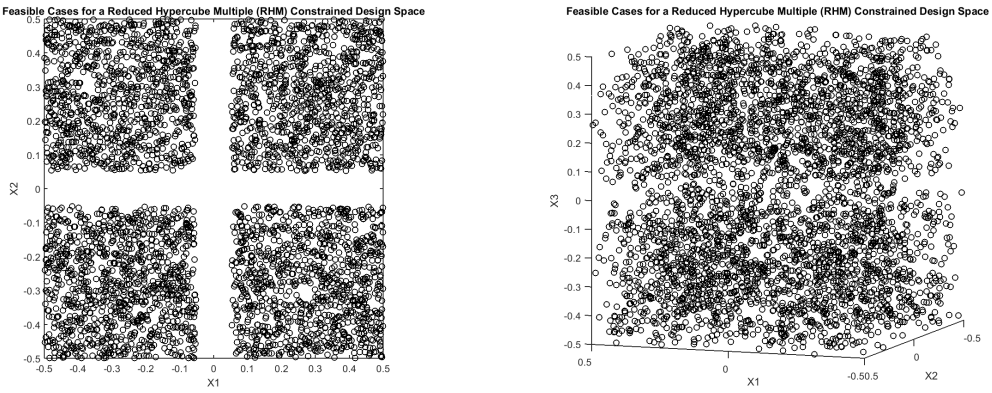

Figure 91: 2-D and 3-D Design Space Visualization for RHM Constrained Design Space

A.1.3 Random Removal Fixed Percentage (RRFP)

This constrained design space had a Fixed Percentage (10 percent) of the cases randomly removed. There are no graphical depictions for this constrained design space

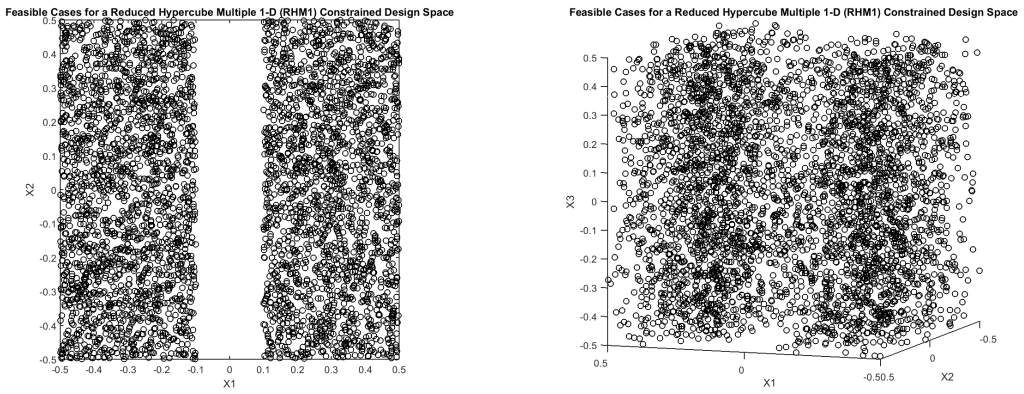

Figure 92: 2-D and 3-D Design Space Visualization for RHM1 Constrained Design Space

as it appears simply like the baseline PMC design space but with fewer cases. In a practical problem a feasible space of this nature could result from random failures throughout the design space.

A.1.3.1 MATLAB code

```
%% Apply Random Removal Fixed Percentage constraint
% (randomly remove 10% of cases)
C_name = 'RRFP';
fprintf('\tConstraint: %s\t', C_name);
tic;
DOE_RRFP = [];
n_RRFP = 0;
sp_RRFP = 1;        % successful case percentage
% Determining the unique indices of the cases to remove
rmv_p = 0.1;     % Percentage of cases to be removed
Ind_vec = sort(ceil(unifrnd(0,n_val,rmv_p*n_val,1)));
unique_check = unique(Ind_vec);
n_needed = length(Ind_vec) - length(unique_check);
while n_needed > 0
```

```
    add_ind = ceil(unifrnd(0,n_val,n_needed,1));
    Ind_vec = [unique_check; add_ind];
    unique_check = unique(Ind_vec);
    n_needed = length(Ind_vec) - length(unique_check);
end
Ind_vec = sort(Ind_vec);
i_Ind = 1;
more_to_add = 0;
for i = 1:n_val
    if i ~= Ind_vec(i_Ind)
        DOE_RRFP = [DOE_RRFP; DOE_base(i,:)];
        n_RRFP = n_RRFP+1;
    else
        i_Ind = i_Ind+1;
        if i_Ind > length(Ind_vec)
            more_to_add = 1;
            break;
        end
    end
end
if more_to_add == 1
    DOE_RRFP = [DOE_RRFP; DOE_base(i+1:n_val,:)];
    n_RRFP = n_RRFP+(n_val-i);
end
sp_RRFP = n_RRFP/n_val;
sp_RRFP_arr(rep_i) = sp_RRFP;
```

A.1.4 Random Removal n/d (RRND)

This constrained design space randomly removed cases until a specified n/d value was achieved. There are no graphical depictions for this constrained design space as it appears simply like the baseline PMC design space but with fewer cases. In a practical problem a feasible space of this nature could result from random failures

throughout the design space.

A.1.4.1 MATLAB code

```
%% Apply Random Removal N/D constraint (randomly remove
% cases until a fixed n/D ratio is hit)
C_name = 'RRND';
fprintf('\tConstraint: %s\t', C_name);
tic;
DOE_RRND = [];
n_RRND = 0;
sp_RRND = 1;       % successful case percentage
n_D = n_val/n_DV_val;
n_D_tgt = n_D_tgt_arr(n_i);
n_to_rmv = 0;
if n_D >= n_D_tgt
    n_to_rmv = (n_D-n_D_tgt)*n_DV_val;
end
Ind_vec = sort(ceil(unifrnd(0,n_val,n_to_rmv,1)));
unique_check = unique(Ind_vec);
n_needed = length(Ind_vec) - length(unique_check);
while n_needed > 0
    add_ind = ceil(unifrnd(0,n_val,n_needed,1));
    Ind_vec = [unique_check; add_ind];
    unique_check = unique(Ind_vec);
    n_needed = length(Ind_vec) - length(unique_check);
end
Ind_vec = sort(Ind_vec);
i_Ind = 1;
more_to_add = 0;
if ~isempty(Ind_vec)
    for i = 1:n_val
```

```
        if i ~= Ind_vec(i_Ind)
            DOE_RRND = [DOE_RRND; DOE_base(i,:)];
            n_RRND = n_RRND+1;
        else
            i_Ind = i_Ind+1;
            if i_Ind > length(Ind_vec)
                more_to_add = 1;
                break;
            end
        end
    end
else
    DOE_RRND = DOE_base;
end
if more_to_add == 1
    DOE_RRND = [DOE_RRND; DOE_base(i+1:n_val,:)];
    n_RRND = n_RRND+(n_val-i);
end
sp_RRND = n_RRND/n_val;
sp_RRND_arr(rep_i) = sp_RRND;
```

A.2 Non-Hypercubic Constrained Design Spaces

The following test problems were representative Non-Hypercubic design spaces featuring cases missing. These were used to test the MI classifier to determine if could detect a non-hypercubic space. Some of these test cases were also utilized to create NHC design spaces for use in Experiments II and III. Each unique constrained design space only features a single type of constraint which determines which cases will remain feasible. In a practical problem, multiple constraints of varying types may be present and thus the global feasible space that results is the intersection of the feasible spaces corresponding to each constraint. In a similar fashion these constrained design spaces can be combined to emulate realistic feasible design spaces that may

be expected from a practical engineering problem.

A.2.1 Hypersphere Removal (HS and HS2)

This constrained design space featured a Hypersphere removed from the cneter of the design space in all dimensions (HS) and only the first 2 dimensions (HS2). The purpose of this constraint was to provide a representative design space that featured a void and thus made the feasible space non-convex and the boundary non-linear.

A.2.1.1 MATLAB code

```
% Hypersphere (HS) -d-ball Void of constant volume
Tgt_Vol = 0.8;    % Change this to change % failed volume
Vball = 1-Tgt_Vol;
% Use different forumla to calculate R based on if n_DV_val is odd or even
R = 0;
if mod(n_DV_val,2) == 0 % n_DV_val is even
    k = (n_DV_val)/2;
    %R = (Vball*factorial(n_DV_val/2)/pi^(n_DV_val/2))^(1/n_DV_val);
    R = (Vball*factorial(k)/pi^(k))^(1/(2*k));
else
    k = (n_DV_val-1)/2;
    R = (Vball*factorial(2*k+1)/(2*factorial(k)*(4*pi)^k))^(1/(2*k+1));
end
n_HS = 0;
DOE_HS = [];
for i = 1:n_val
    r_i = sqrt(sumsqr(DOE_base(i,:))); % For MATLAB
    %r_i = sqrt(sumsq(DOE_base(i,:))); % For OCTAVE
    if r_i >= R
        DOE_HS = [DOE_HS; DOE_base(i,:)];
        n_HS = n_HS+1;
```

```
    end
end
```

A.2.1.2 Graphical Representation

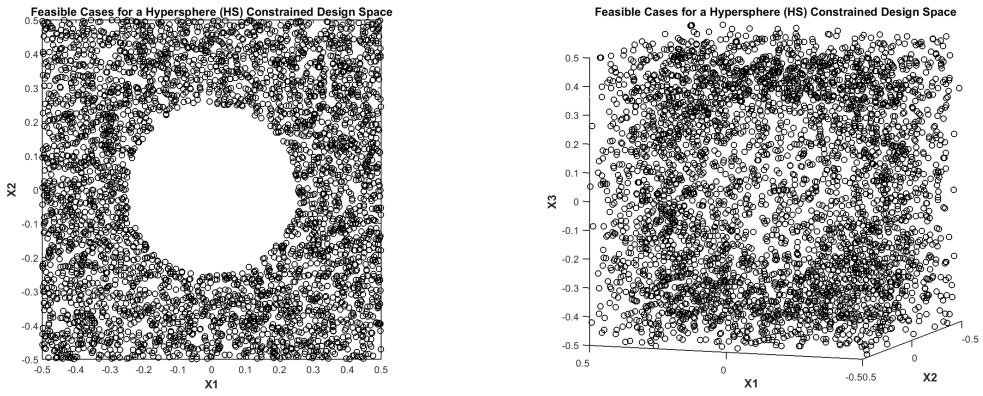

Figure 93: 2-D and 3-D Design Space Visualization for HS Constrained Design Space

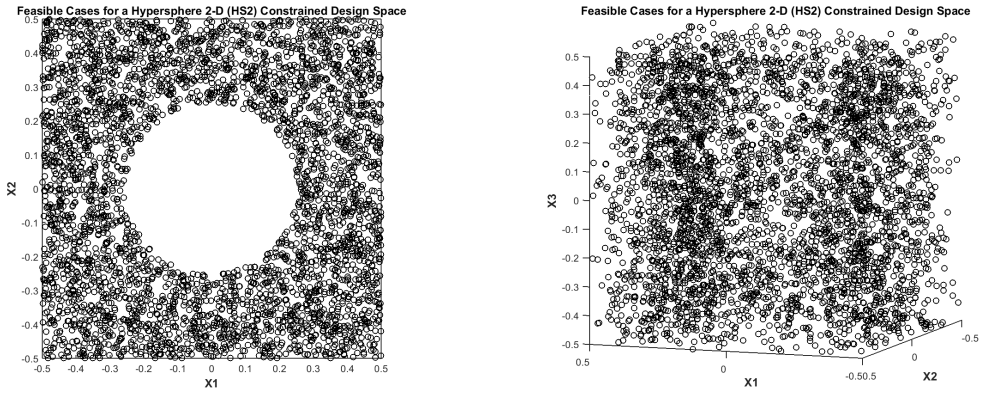

Figure 94: 2-D and 3-D Design Space Visualization for HS2 Constrained Design Space

A.2.2 Linear Constraint Large (LCL and LCL2)

This constrained design space featured a Large Linear Constraint in a corner of the design space in all dimensions (LCL) and only the first 2 dimensions (LCL2). The purpose of this constraint was to provide a representative design space that featured

a linear constraint to make the feasible space NHC. As it was a 'large' constraint it was meant to be easier to detect and/or successfully bound.

A.2.2.1 MATLAB code

```
% Linear Constraint Large
Tgt_Vol = 0.2;    % Change this to change % failed volume
max_bound = 0.5;
base_len = (Tgt_Vol*factorial(n_DV_val))^(1/n_DV_val);
n_LCL = 0;
DOE_LCL = [];
DOE_F = [];
for i = 1:n_val
    if sum(DOE_base(i,:)) <= (max_bound*n_DV_val - base_len)
        DOE_LCL = [DOE_LCL; DOE_base(i,:)];
        n_LCL = n_LCL+1;
    else
        DOE_F = [DOE_F; DOE_base(i,:)];
    end
end

% Linear Constraint Large 2-D
Tgt_Vol = 0.2;    % Change this to change % failed volume
max_bound = 0.5;
base_len = (Tgt_Vol*factorial(2))^(1/2);
n_LCL2 = 0;
DOE_LCL2 = [];
DOE_F = [];
for i = 1:n_val
    if sum(DOE_base(i,1:2)) <= (max_bound*2 - base_len)
        DOE_LCL2 = [DOE_LCL2; DOE_base(i,:)];
        n_LCL2 = n_LCL2+1;
```

```
    else
        DOE_F = [DOE_F; DOE_base(i,:)];
    end
end
```

A.2.2.2 Graphical Representation

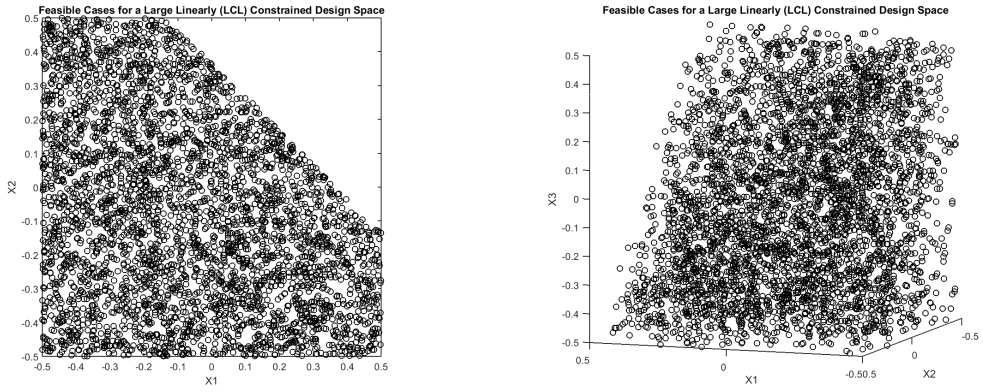

Figure 95: 2-D and 3-D Design Space Visualization for LCL Constrained Design Space

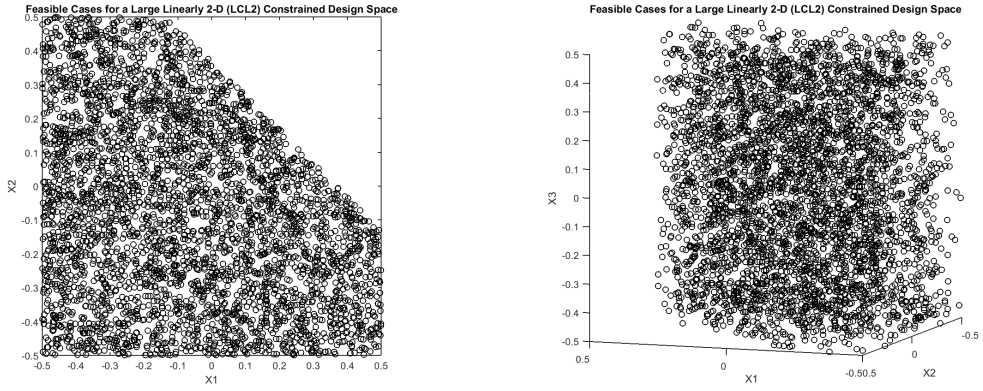

Figure 96: 2-D and 3-D Design Space Visualization for LCL2 Constrained Design Space

A.2.3 Linear Constraint Small (LCS and LCS2)

This constrained design space featured a Small Linear Constraint in a corner of the design space in all dimensions (LCS) and only the first 2 dimensions (LCS2). The

purpose of this constraint was to provide a representative design space that featured a linear constraint to make the feasible space NHC. As it was a 'small' constraint it was meant to be harder to detect and/or successfully bound.

A.2.3.1 MATLAB code

```
% Linear Constraint Small
% Code is same as LCL but with Tgt_Vol = 0.05 instead of 0.2
Tgt_Vol = 0.05;    % Change this to change % failed volume
```

A.2.3.2 Graphical Representation

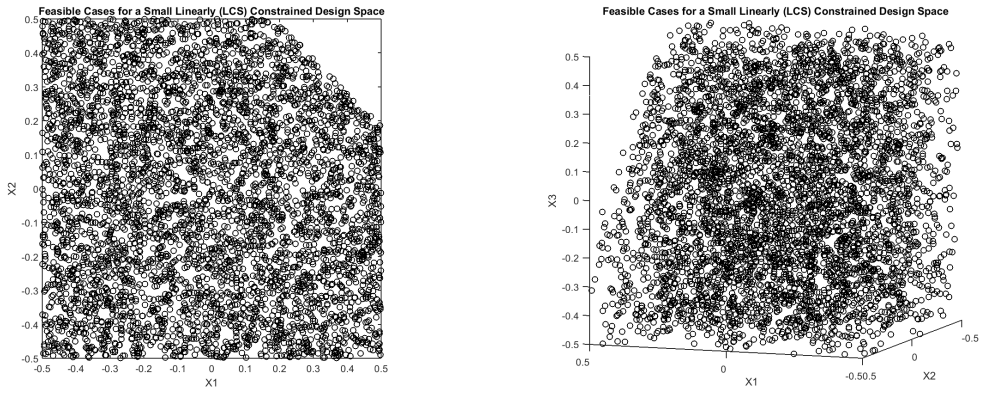

Figure 97: 2-D and 3-D Design Space Visualization for LCS Constrained Design Space

A.2.4 Non-Linear Constraint Large (NLCL and NLCL2)

This constrained design space featured a Large Non-Linear Constraint in a corner of the design space in all dimensions (NLCL) and only the first 2 dimensions (NLCL2). The purpose of this constraint was to provide a representative design space that featured a linear constraint to make the feasible space NHC and the boundary non-convex. As it was a 'large' constraint it was meant to be easier to detect and/or successfully bound.

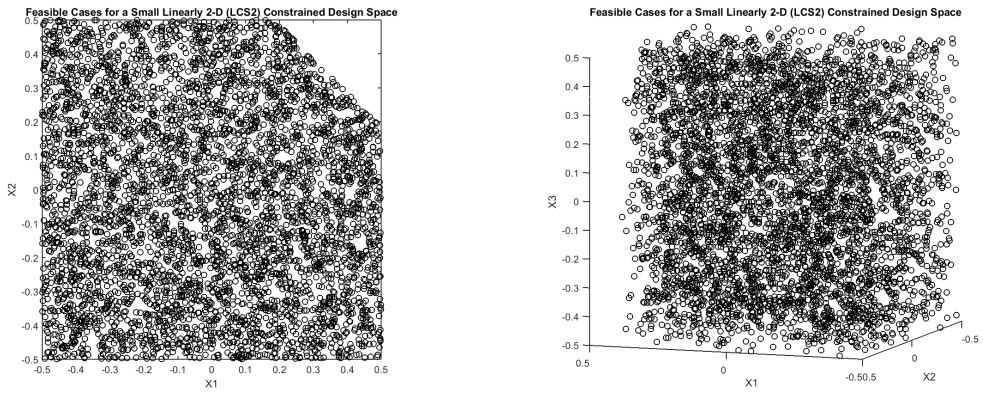

Figure 98: 2-D and 3-D Design Space Visualization for LCS2 Constrained Design Space

A.2.4.1 MATLAB code

```
% Non-Linear Constraint
Tgt_Vol = 0.2;    % Change this to change % failed volume
Vball = 2^n_DV_val*Tgt_Vol;
max_bound = 0.5;
% Use different forumla to calculate R based on if n_DV_val is odd or even
R = 0;
if mod(n_DV_val,2) == 0 % n_DV_val is even
    k = (n_DV_val)/2;
    %R = (Vball*factorial(n_DV_val/2)/pi^(n_DV_val/2))^(1/n_DV_val);
    R = (Vball*factorial(k)/pi^(k))^(1/(2*k));
else
    k = (n_DV_val-1)/2;
    R = (Vball*factorial(2*k+1)/(2*factorial(k)*(4*pi)^k))^(1/(2*k+1));
end
n_NLCL = 0;
DOE_NLCL = [];
for i = 1:n_val
    case_translated = max_bound - DOE_base(i,:);
```

```
    r_i = sqrt(sumsqr(case_translated)); % For MATLAB
    %r_i = sqrt(sumsq(case_translated)); % For OCTAVE
    if r_i >= R
        DOE_NLCL = [DOE_NLCL; DOE_base(i,:)];
        n_NLCL = n_NLCL+1;
    end
end

% Non-Linear Constraint 2-D
Tgt_Vol = 0.2;    % Change this to change % failed volume
Vball = 2^2*Tgt_Vol;
max_bound = 0.5;
% Use different forumla to calculate R based on if n_DV_val is odd or even
R = 0;
k = 1;  % k = 2/2
R = (Vball*factorial(k)/pi^(k))^(1/(2*k));
n_NLCL2 = 0;
DOE_NLCL2 = [];
for i = 1:n_val
    case_translated = max_bound - DOE_base(i,1:2);
    r_i = sqrt(sumsqr(case_translated)); % For MATLAB
    %r_i = sqrt(sumsq(case_translated)); % For OCTAVE
    if r_i >= R
        DOE_NLCL2 = [DOE_NLCL2; DOE_base(i,:)];
        n_NLCL2 = n_NLCL2+1;
    end
end
```

A.2.4.2 Graphical Representation

A.2.5 Non-Linear Constraint Small (NLCS and NLCS2)

This constrained design space featured a Small Non-Linear Constraint in a corner of the design space in all dimensions (NLCS) and only the first 2 dimensions (NLCS2).

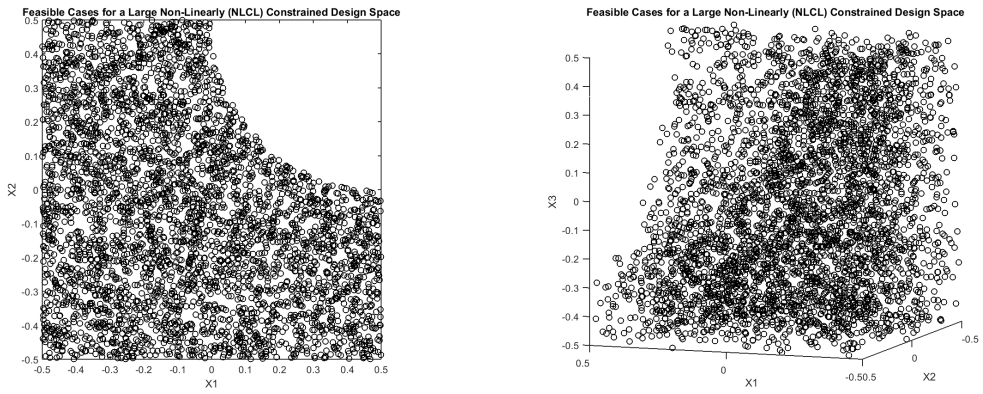

Figure 99: 2-D and 3-D Design Space Visualization for NLCL Constrained Design Space

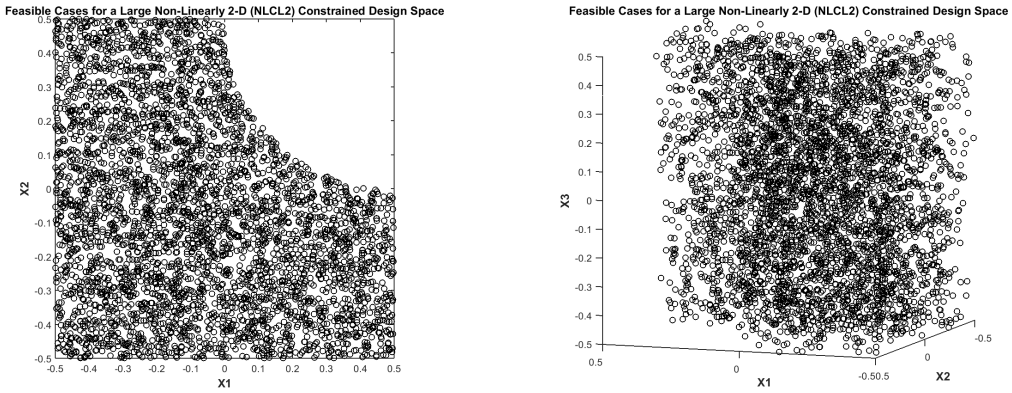

Figure 100: 2-D and 3-D Design Space Visualization for NLCL2 Constrained Design Space

The purpose of this constraint was to provide a representative design space that featured a non-linear constraint to make the feasible space NHC and the boundary non-convex. As it was a 'small' constraint it was meant to be harder to detect and/or successfully bound.

A.2.5.1 MATLAB code

```
% Non-Linear Constraint Small
```

```
% Code is same as NLCL but with Tgt_Vol = 0.05 instead of 0.2
Tgt_Vol = 0.05;    % Change this to change % failed volume
```

A.2.5.2 Graphical Representation

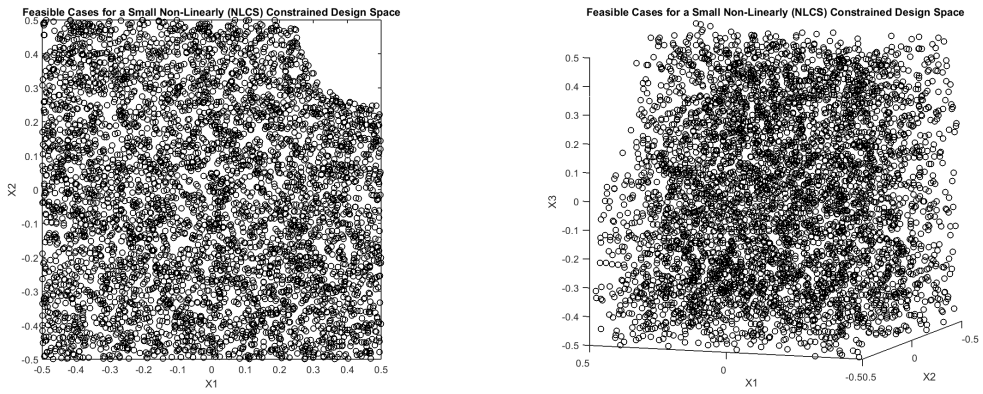

Figure 101: 2-D and 3-D Design Space Visualization for NLCS Constrained Design Space

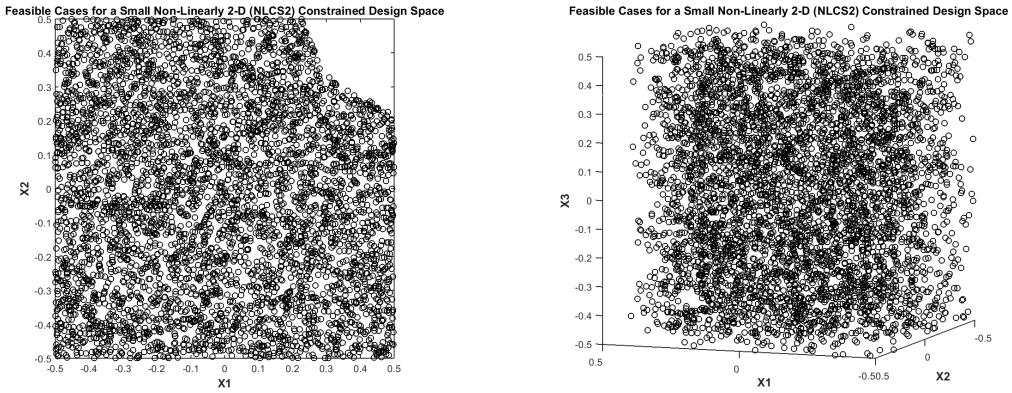

Figure 102: 2-D and 3-D Design Space Visualization for NLCS2 Constrained Design Space

A.2.6 Checkerboard Coarse (CBC and CBC2)

This constrained design space featured a Coarse Checkerboard Constraint (2 bins per dimension) affecting the design space in all dimensions (CBC) and only the first 2 dimensions (CBC2). The purpose of this constraint was to provide a representative

design space that featured strong correlation between variables and a discontinuous feasible space. This constraint denied half of the design space and due to its 'coarse' pattern it was meant to be easier to detect and/or successfully bound. Although this constraint would be unlikely to appear in this form in a practical design space, it could be representative of discontinuous plateaus or point clouds that can appear in feasible spaces once performance constraints are applied. Additionally this constrained design space has been featured in literature for use in demonstrating MI's ability to detect correlation in various design spaces [45].

A.2.6.1 MATLAB code

```
% Checkerboard Coarse in all dims
n_bins = 2;
bin_rng = 1/n_bins;
DOE_binned = ceil((DOE_base+0.5).*n_bins);
DOE_sign = -1.*mod(DOE_binned,2)+0.5;
DOE_prod = prod(transpose(DOE_sign));
n_CBC = 0;
DOE_CBC = [];
for i = 1:n_val
    if DOE_prod(i) > 0
        DOE_CBC = [DOE_CBC; DOE_base(i,:)];
        n_CBC = n_CBC+1;
    end
end

% Checkerboard Coarse only in first 2 dims
DOE_base = unifrnd(-0.5,0.5,n_val,n_DV_val);
n_bins = 2;
bin_rng = 1/n_bins;
```

```
DOE_binned = ceil((DOE_base(:,1:2)+0.5).*n_bins);
DOE_sign = -1.*mod(DOE_binned,2)+0.5;
DOE_prod = prod(transpose(DOE_sign));
n_CBC2 = 0;
DOE_CBC2 = [];
num_Constraints = 1;
Constraint_mat = zeros(n_val,num_Constraints);
for i = 1:n_val
    if DOE_prod(i) > 0
        DOE_CBC2 = [DOE_CBC2; DOE_base(i,:)];
        n_CBC2 = n_CBC2+1;
        Constraint_mat(i) = 1;
    end
end
```

A.2.6.2 Graphical Representation

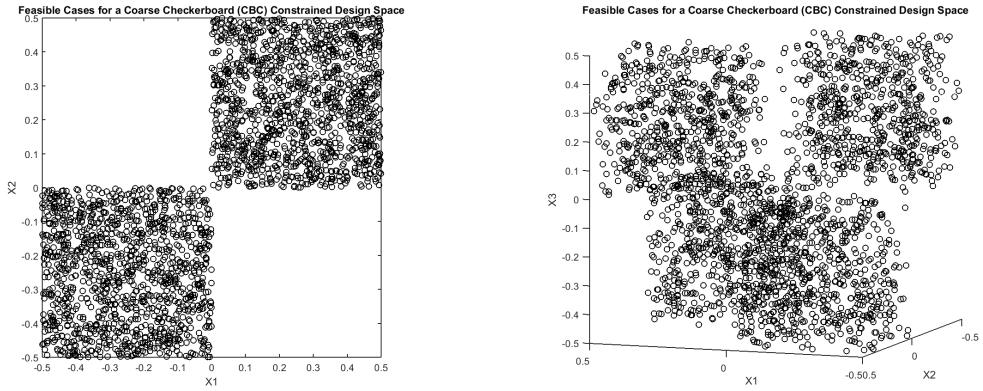

Figure 103: 2-D and 3-D Design Space Visualization for CBC Constrained Design Space

A.2.7 Checkerboard Fine (CBF and CBF2)

This constrained design space featured a Fine Checkerboard Constraint (10 bins per dimension) affecting the design space in all dimensions (CBF) and only the first 2 dimensions (CBF2). The purpose of this constraint was to provide a representative

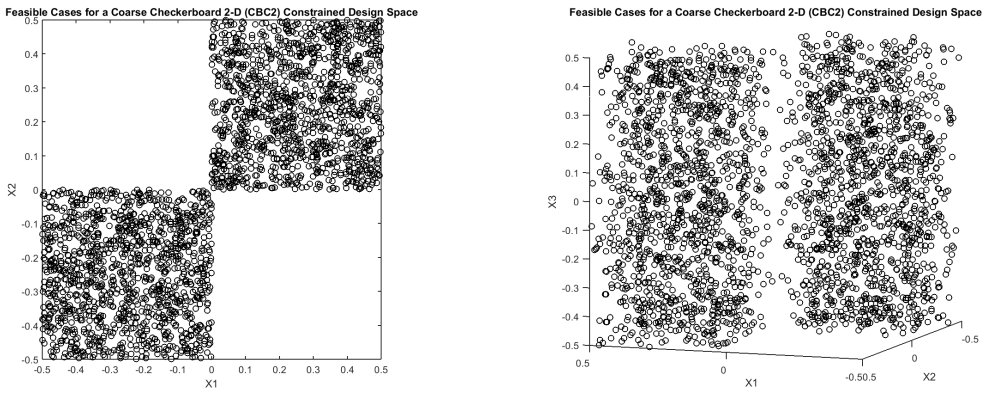

Figure 104: 2-D and 3-D Design Space Visualization for CBC2 Constrained Design Space

design space that featured strong correlation between variables and a discontinuous feasible space. This constraint denied half of the design space and due to its 'fine' pattern it was meant to be harder to detect and/or successfully bound. Again, this type of constraint would likely not manifest in this form in a practical problem, but could be representative for design spaces subjected to pockets of infeasibility due to convergence issues.

A.2.7.1 MATLAB code

```
% Checkerboard Fine in all dims
% Code is same as CBC except n_bins = 10 instead of 2
n_bins = 10;
```

A.2.7.2 Graphical Representation

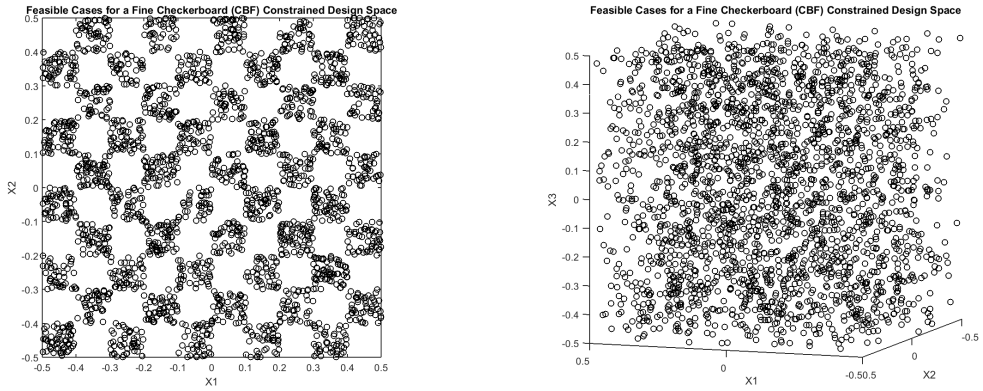

Figure 105: 2-D and 3-D Design Space Visualization for CBF Constrained Design Space

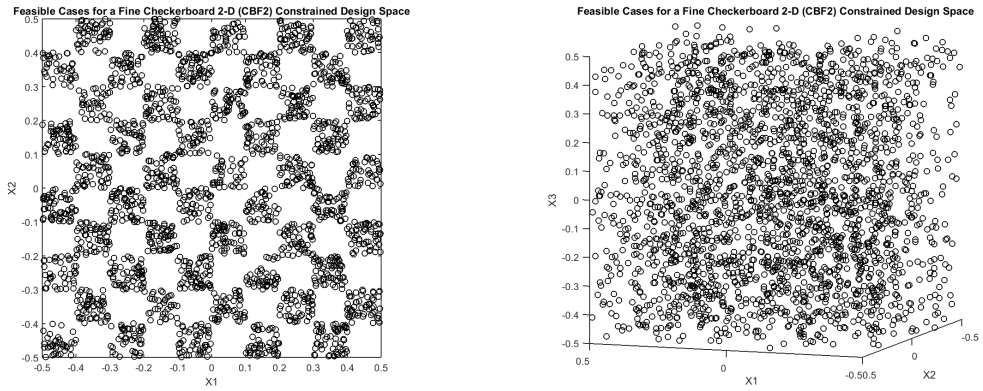

Figure 106: 2-D and 3-D Design Space Visualization for CBF2 Constrained Design Space

APPENDIX B

EDS LTA HWB DESIGN SPACE DETAILS

The following contains details of the variables and constraints/failure modes pertinent to the LTA HWB design spaces investigated with the EDS Modeling and Simulation Environment in Experiment IV.

B.1 97-D LTA HWB Design Space

The 97-D LTA HWB Design Space (and the output from its DSE) was not constructed by the author but rather post processed. As such the design variables used for the modeling of the LTA HWB are a collection of vehicle parameters and technology k-factors (scalars on model parameters to account for the influences of technology) which were deemed relevant by users of EDS for the modeling of a technology infused LTA HWB aircraft concept. The ranges on the variables shown have been refined by expert users through years of modeling aircraft concepts within EDS. Thus the 97-D design space, while it does feature failed/infeasible designs, has been tuned in the DV to bound a hypervolume of interest with an acceptable percentage of failed designs.

B.1.1 Design Variables

The following table lists the details of the 97 continuous design variables used to define this particular design space for the LTA HWB.

Table 14: Design Variable Details (97-D 1 of 6)

Variable	Lower Limit	Upper Limit	Parent Code	Description
ABTC	0.000	0.010	NPSS	Bleed flow required for ABTC
Burner_Liner_rho	0.069	0.322	WATE	Burner liner material density
Burnereff	0.992	0.997	NPSS	Burner efficiency
Byp_Nozz_s_Wt	0.850	1.500	WATE	Bypass nozzle weight scalar
Cooled Cooling Nondimensional Weight	0.000	9.000	NPSS	Cooled cooling nondimensional weight
Core_Nozz_s_Wt	0.650	1.150	WATE	Core nozzle weight scalar
Cust_Bleed	0.590	3.930	NPSS	Engine customer bleed (customer)
Cust_Bleed_Map	0.000	0.040	NPSS	Engine customer bleed (function of ambient)
d_Burn_dP	0.040	0.050	NPSS	Burner pressure drop intercept
DISAP	-10.115	0.000	ANOPP	Suppression factor on fan discharge noise (Approach)
DISTO	-11.740	0.000	ANOPP	Suppression factor on fan discharge noise (Takeoff)
Duct15_dP	0.018	0.028	NPSS	Duct 15 pressure drop (bypass duct)
Ext_Ratio	1.000	1.300	NPSS	Extraction ratio at Aero Design Point
Fan_AR_Fact	1.000	1.500	WATE	Aspect ratio factor applied to fan blades and stators
Fan_bladeSolidity	0.060	1.500	WATE	Fan blade solidity
Fan_Deff	-0.018	0.013	NPSS	Fan efficiency delta at Aero Design Point (from historical curve)
Fan_HtoT	0.250	0.300	NPSS	Fan hub to tip ratio

Table 15: Design Variable Details (97-D 2 of 6)

Variable	Lower Limit	Upper Limit	Parent Code	Description
Fan_numBlades	16.000	18.000	WATE	Number of fan blades
Fan_SpecW	42.752	45.270	NPSS	Fan specific flow at Aero Design Point
FCDO	0.989	1.000	FLOPS	Lift independent drag factor
FCDSUB	0.900	1.100	FLOPS	Factor to increase or decrease all subsonic drag coefficients
FPR	1.300	1.500	NPSS	Fan Pressure Ratio at Aero Design Point
FRFU	0.680	1.000	FLOPS	Fuselage weight factor
FRLGM	1.000	1.030	FLOPS	Landing gear weight, main
FRLGN	1.000	1.030	FLOPS	Landing gear weight, nose
FRWI	0.750	1.000	FLOPS	total wing weight factor
FRWI1	1.165	1.576	FLOPS	First term in wing weight equation- loosely corresponds to bending material weight
FRWI2	0.572	0.783	FLOPS	Second term in wing weight equation- loosely corresponds to control surfaces, spars and ribs
GearBoxLosses	0.010	0.015	NPSS	Percent losses from gearbox-Applied to LP shaft
GustLoad	0.600	1.100	WATE	Gust load sizing load as a percent of normal
GW	4.932e5	5.548e5	FLOPS	Ramp weight, lb-Initial guess
HPC_AFC_LossRatio	1.000	1.500	NPSS	Ratio of baseline loss coefficient over loss coefficient with endwall and boundary layer active flow control
HPC_AFC_nStages	0.000	4.000	NPSS	Number of HPC stages to apply AFC efficiency gain

Table 16: Design Variable Details (97-D 3 of 6)

Variable	Lower Limit	Upper Limit	Parent Code	Description
HPC_Deff	0.004	0.050	NPSS	HPC efficiency delta at aero design point
HPC_Disk_rho	0.800	1.000	WATE	HPC disk material density
HPC_Dutip	-380.000	-150.074	NPSS	HPC tip speed delta at aero design point
HPC_FlowControl	0.000	0.020	NPSS	Bleed ow required per stage
HPC_FSPRmax	1.660	1.900	NPSS	Maximum HPC 1st stage PR
HPCPR	16.000	22.000	NPSS	HPCPR at aero design point
HPT_Blade_rho	0.302	0.312	WATE	HPT blade material density
HPT_delta_ desBladeTemp	50.000	450.000	NPSS	HPT blade temperature increase
HPT_delta_ desVaneTemp1	50.000	675.000	NPSS	HPT vane 1 temperature increase
HPT_delta_ desVaneTemp2	50.000	675.000	NPSS	HPT vane 2 temperature increase
HPT_eff	0.925	0.946	NPSS	HPT adiabatic efficiency at Aero Design Point
HPT_Load	0.930	1.230	WATE	HPT GE loading
HPT_Stator_rho	0.180	0.312	WATE	HPT stator material density
HPX	0.000	457.000	NPSS	Engine horse power extraction (constant)
HPX_map_highAlt	0.000	250.000	NPSS	Engine horse power extraction needed above 18000k (function of ambient)
HX_deltaT	0.000	400.000	NPSS	Cooled cooling ow temperature drop across heat exchanger
HX_effect	0.700	0.900	NPSS	Cooled Cooling heat exchanger effectiveness
INLAP	-12.885	0.000	ANOPP	Suppression factor on inlet noise

Table 17: Design Variable Details (97-D 4 of 6)

Variable	Lower Limit	Upper Limit	Parent Code	Description
INLTO	-12.880	0.000	ANOPP	Suppression factor on inlet noise
IntercoolerBleedFlow	0.000	0.080	NPSS	Intercooler bleed flow from bypass
IntercoolerCoreDP	0.000	0.050	NPSS	Intercooler core stream dP
IntercoolerHX_effect	0.700	0.900	NPSS	Intercooler heat exchanger effectiveness
Intercooler Nondimensional Weight	0.000	11.000	NPSS	Intercooler non-dimensional weight
JETTO	-3.625	0.000	ANOPP	Suppression factor on jet noise
LPC_Deff	0.018	0.058	NPSS	LPC efficiency delta at Aero Design Point
LPC_Disk_rho	0.800	1.000	WATE	LPC Disk material density
LPC_Stator_rho	0.052	0.168	WATE	LPC stator material density
LPCPR	1.364	3.125	NPSS	LPCPR at aero design point
LPT_AFC_LossRatio	1.000	1.500	WATE	Ratio of baseline loss coefficient over loss coefficient with active flow control
LPT_AFC_nStages	0.000	2.000	NPSS	Number of rear LPT stages to apply AFC
LPT_Blade_rho	0.157	0.313	WATE	LPT blade material density
LPT_Blade2_rho	0.157	0.286	WATE	LPT blade material density
LPT_Deff	0.020	0.112	NPSS	LPT efficiency delta
LPT_delta_ desBladeTemp	50.000	450.000	NPSS	LPT blade temperature increase
LPT_delta_ desVaneTemp	50.000	675.000	NPSS	LPT vane temperature increase
LPT_FlowControl	0.000	0.015	NPSS	Bleed flow required for LPT ow control
LPT_Load	1.000	1.300	WATE	LPT GE loading

Table 18: Design Variable Details (97-D 5 of 6)

Variable	Lower Limit	Upper Limit	Parent Code	Description
LPT_Stator_rho	0.157	0.313	WATE	LPT stator material density
LPT_Stator2_rho	0.157	0.286	WATE	LPT stator material density
MGRAP	-3.000	0.000	ANOPP	Suppression factor on main landing gear
NGRAP	-3.000	0.000	ANOPP	Suppression factor on nose landing gear
PER1	1.250	1.350	ANOPP	Core nozzle chevrons 1=no chevrons, 2= full coverage chevrons
PER2	1.000	1.350	ANOPP	Bypass nozzle chevrons 1=no chevrons, 2= full coverage chevrons
S_BypNozzCv_lowAlt	0.990	1.000	NPSS	Core nozzle velocity coefficient scalar at low altitude
s_CDft_wing	0.940	1.000	NPSS	Scalar for the turbulent skin friction drag on the wing
S_CoreNozzCv_highAlt	0.995	1.000	NPSS	Core nozzle velocity coefficient scalar at high altitude
S_CoreNozzCv_lowAlt	0.990	0.998	NPSS	Core nozzle velocity coefficient scalar at low altitude
s_HPT_ChargeEff	0.650	1.000	NPSS	HPT chargeable (exit) cooling effectiveness factor scalar
s_HPT_NonChargeEff	0.900	1.000	NPSS	HPT non-chargeable (inlet) cooling effectiveness factor scalar
sAccess_Wt	0.072	0.172	WATE	Engine accessories weight fraction of bare engine weight
sInl_Nacelle_thick	0.340	1.000	WATE	Nacelle radius delta scalar
SWETF	0.940	1.000	ANOPP	Fuselage wetted area scalar
T4margin	-171.000	-121.400	NPSS	Difference in T4 between MTO and MCT

Table 19: Design Variable Details (97-D 6 of 6)

Variable	Lower Limit	Upper Limit	Parent Code	Description
T4max	3300.000	3700.000	NPSS	Maximum T4 (set at Take off)
ThrustReverserWeight	0.500	1.000	WATE	Scalar for thrust reverser weight
TOC_Wratio	1.019	1.041	NPSS	Mass flow of Top of Climb to Aero Design Poin
TransREWingUpper	0.000	24.000	NPSS	Turbulent transition Reynolds number for upper wing surface assuming a 20 degree sweep
TRLN	0.000	50.000	FLOPS	Percent laminar flow nacelle lower surface
TRUN	0.000	50.000	FLOPS	Percent laminar flow nacelle upper surface
TWR	0.257	0.284	FLOPS	Thrust to weight ratio- (DESIGN)
VCTE	0.000	0.500	NPSS	Variable camber trailing edge scalar
WAC	1.000	2.200	FLOPS	Air conditioning group weight scalar
WAPU	1.000	3.000	FLOPS	Auxiliary power unit weight scalar
WHYD	0.800	1.000	FLOPS	Hydraulics group weight scalar

B.1.2 Constraints/Failure Modes

The 97-D LTA HWB design space was subject to nine distinct modes of failure/infeasibility. These modes all represent code-based or convergence failures encountered during exploration of the 97-D design space. It is important to note that none of

these failure modes were specified a-priori nor do they represent any form of performance constraint (they are native to the Modeling and Simulation environment and HTCondor). The failure modes encountered for the 97-D design space were as follows:

- **FLOPS-ZFW**: FLOPS Zero Fuel Weight ZFW error (vehicle sizing) - occurs when the sized vehicle returns a zero fuel weight before the mission has been completed
- **ANOPP**: ANOPP error (noise) - failure of the noise module to run, one of the last errors encountered in the model execution chain
- **CONDOR**: HTCondor error (distributed computing) - failure of the distributed computing software or remote host to properly run and return a job
- **Cum-Noise**: Cumulative Noise below threshold (noise) - failure at one or more of the certification noise points such that the cumulative aircraft noise value calculated is unrealistic
- **ROC**: Rate Of Climb insufficient error (vehicle sizing) - failure in the vehicle sizing routine in which the aircraft does not maintain the capability for a positive rate of climb throughout its mission profile.
- **MDP**: Multi-Design Point error (engine sizing) - failure to converge on an engine design which could simultaneously satisfy the constraints at the multiple design points (Takeoff, Top Of Climb, etc.)
- **Thrust-Conv**: Thrust Convergence error (engine sizing) - engine sizing could not converge on the required thrust within a set number of iterations
- **WATE**: WATE error (engine flowpath) - failure in the designed engine flowpath to meet requirements

- **Main-Conv**: Main Design Loop Convergence error (convergence) - design loop between between engine and vehicle sizing could not converge within a set number of iterations

B.2 50-D LTA HWB Design Space

The 50-D LTA HWB Design Space (and the output from its DSE) was constructed and run by the author. In order to increase the available resolution for a given case budget, the number of design variables was reduced from 97 to 50, while retaining the most important variables of the original set. An additional variable (TO_Thrust) was added to the original set, but the ranges on the variables remained unchanged except for a single variable, TWR.

B.2.1 Design Variables

As the 50-D design space was for the most part a pared down version of the 97-D space only the differences between the two will be shown in this section. The following variables were kept for the 50-D space and their ranges perserved:

HPC_Dutip, HPC_FSPRmax, HPCPR, IntercoolerHX_effect, LPCPR, Burnereff, HX_deltaT, Cust_Bleed_Map, d_Burn_dP, Duct15_dP, Fan_Deff, HPC_Deff, HPT_delta_des BladeTemp, HPT_delta_des VaneTemp1, HPT_delta_des VaneTemp2, HPT_eff, HPT_Load, HPX, HPX_map_highAlt, Intercooler BleedFlow, Intercooler CoreDP, LPC_Deff, LPT_Deff, LPT_delta_des BladeTemp, LPT_delta_desVaneTemp, LPT_Load, T4margin, TRLN, TRUN, WAC, Byp_Nozz_s_Wt, FCDO, FCDSUB, FRFU, FRWI, FRWI1, FRWI2, s_HPT_ChargeEff, s_HPT_NonChargeEff, GW, TO_Thrust, TWR, GustLoad, VCTE, s_CDft_wing, sInl_Nacelle_thick, CooledCooling NondimensionalWeight, Intercooler NondimensionalWeight, TransREWingUpper, SWETF

The following table lists the differences between the 97-D space for two of the 50 continuous design variables used to define this particular design space for the LTA HWB.

Table 20: Design Variable Details (50-D Differences)

Variable	Lower Limit	Upper Limit	Parent Code	Description
TO_Thrust	4.973e4	6.728e4	NPSS	Takeoff thrust [lbs]
TWR	0.210	0.284	FLOPS	Thrust to weight ratio- (DESIGN)

B.2.2 Constraints/Failure Modes

The 50-D LTA HWB design space was subject to seven distinct modes of failure/infeasibility. These modes all represent code-based or convergence failures encountered during exploration of the 50-D design space. The noise failure modes are missing for this design space as the noise module was deactivated for DSE of this design space. It is important to note that none of these failure modes were specified a-priori nor do they represent any form of performance constraint (they are native to the Modeling and Simulation environment and HTCondor). The failure modes encountered for the 50-D design space were as follows:

- **FLOPS-ZFW**: FLOPS Zero Fuel Weight ZFW error (vehicle sizing) - occurs when the sized vehicle returns a zero fuel weight before the mission has been completed
- **CONDOR**: HTCondor error (distributed computing) - failure of the distributed computing software or remote host to properly run and return a job
- **ROC**: Rate Of Climb insufficient error (vehicle sizing) - failure in the vehicle sizing routine in which the aircraft does not maintain the capability for a positive rate of climb throughout its mission profile.
- **MDP**: Multi-Design Point error (engine sizing) - failure to converge on an engine design which could simultaneously satisfy the constraints at the multiple

design points (Takeoff, Top Of Climb, etc.)

- **Thrust-Conv**: Thrust Convergence error (engine sizing) - engine sizing could not converge on the required thrust within a set number of iterations
- **NPSS-OFF**: NPSS Off-Design error (engine off-design) - failure in the designed engine to meet requirements of off design conditions
- **Main-Conv**: Main Design Loop Convergence error (convergence) - design loop between between engine and vehicle sizing could not converge within a set number of iterations

REFERENCES

[1] "High-Level Meeting on International Aviation and Climate Change: A Global Sectoral Approach for Aviation." 2009.

[2] "High-Level Meeting on International Aviation and Climate Change: Provisional Agenda and Annotations." 2009.

[3] Alexandros Karatzoglou, Alex Smola, Kurt Hornik, *Kernel-Based Machine Learning Lab*. R Foundation for Statistical Computing, 2016.

[4] Alexandrov, N., H. M., *Multidisciplinary design optimization : State of the art : [proceedings of the ICASE/NASA Langley Workshop on Multidisciplinary Design Optimization, Hampton, Virginia, March 13-16, 1995]*. Philadelphia: SIAM, 1997.

[5] Anderson, J. D., *Modern Compressible Flow With Historical Perspective*. McGraw Hill, 3rd ed., 2003.

[6] Antony, J., *Design of experiments for engineers and scientists*. 2014.

[7] Astakhov, S.A., S. H. K. A. and Grassberger, P., "Spectral mixture decomposition by least dependent component analysis," 2004.

[8] Astakhov, S.A., S. H. K. A. and Grassberger, P., "Monte carlo algorithm for least dependent non-negative mixture decomposition," *Analytical Chemistry*, vol. 78, no. 5, pp. 1620–1627, 2006.

[9] Barber, D., *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 1st ed., 2012.

[10] Bartholomew, P. and Wellen, H. K., "Computer-aided optimization of aircraft structures," *Journal of Aircraft*, vol. 27, no. 12, pp. 1079–1086, 1990.

[11] Berkes, U. L., "Efficient optimization of aircraft structures with a large number of design variables," *Journal of Aircraft*, vol. 27, no. 12, pp. 1073–1078, 1990.

[12] Bertin, J. J. and Cummings, R. M., *Aerodynamics for Engineers*. Pearson Prentice-Hall, 5th ed., 2009.

[13] Bishop, C. M., *Pattern Recognition and Machine Learning*. Springer, 1st ed., 2006.

[14] Breiman, L., "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[15] Cox, D. and Reid, N., *The Theory of Design of Experiments*. Chapman and Hall/CRC, 1st ed., 2000.

[16] de Berg, M., van Kreveld, M., Overmars, M., and Schwarzkopf, O., *Computational Geometry: Algorithms and Applications*. Springer-Verlag, 2nd ed., 2000.

[17] Denney, R. K., T. J. C. and Mavris, D. N., "Emissions Prediction for Aircraft Conceptual Design," in *48th AIAA-ASME-SAE-ASEE Joint Propulsion Conference and Exhibit*, (Atlanta, GA), August 2012.

[18] Dodd, A. J., K. K. E. L. M. J. R. B. A. S. G. D. S. R. C. and Tzong, T. J., "Aeroelastic design optimization program," *Journal of Aircraft*, vol. 27, no. 12, pp. 1028–1036, 1990.

[19] Drela, M. and Youngren, H., *AVL 3.30 User Primer*. MIT Aero & Astro, August 2010.

[20] Dudley, J. M., H. X. H. R. T. G. B. and Mason, W. H., " Variable Complexity Interlacing of Weight Equations and Structural Optimization for the Design of the High Speed Civil Transport," in *Proceedings of the 5th AIAA Multidisciplinary Analysis and Optimization Symposium*, (Panama City, FL), AIAA Paper 94-4377-CP, September 1994.

[21] Fernandez-Delgado, M., C. E. B. S. A. D., "Do we need hundreds of classifiers to solve real world classification problems?," *Journal of Machine Learning Research*, vol. 15, pp. 3133–3183, 2014.

[22] Fisher, R. A., *The Design of Experiments*. Oliver and Boyd, 8th ed., 1966.

[23] Fortran original by Leo Breiman and Adele Cutler, R port by Andy Liaw and Matthew Wiener, *Breiman and Cutler's Random Forests for Classication and Regression*. R Foundation for Statistical Computing, 2015.

[24] Gallman, J. W., K. R. W. C. R. M. and Hinson, M. L., "Optimization of an Advanced Business Jet," in *Proceedings of the 5th AIAA Multidisciplinary Analysis and Optimization Symposium*, (Panama City, FL), pp. 482–492, AIAA Paper 94-4303-CP, 1994.

[25] Gates, M. and Lewis, M., "Optimization of Spacecraft Orbit and Shielding for Radiation Dose," in *Proceedings of the 4th AIAA/NASA/USAF/OAI Symposium on Multidisciplinary Analysis and Optimization*, (Cleveland, OH), AIAA Paper No. 92-4771, September 1992.

[26] Gatian, K., *A Quantitative Model-Driven Approach to Technology Selection and Development Through Epistemic Uncertainty Reduction*. PhD thesis, Georgia Institute of Technology, 2015.

[27] Gauch Jr., H., "Winning the accuracy game," *American Scientist*, vol. 94, no. 2, pp. 133–141, 2006.

[28] Gere, J. M., *Mechanics of Materials*. Thomson-Engineering, 6th ed., 2003.

[29] Giunta, A., Wojtkiewicz, S., and Eldred, M., "Overview of modern design of experiments methods for computational simulations," *Proceedings of the 41st AIAA ...*, no. January, pp. 1–17, 2003.

[30] Gladin, J. C., S. J. S. K. B. K. and Mavris, D. N., "Effects of Boundary Layer Ingesting (BLI) Propulsion Systems on Engine Cycle Selection and HWB Vehicle Sizing," in *48th AIAA-ASME-SAE-ASEE Joint Propulsion Conference and Exhibit*, (Atlanta, GA), 2012.

[31] Gray, A., *Enhancement of Set-Based Design Practices Via Introduction of Uncertainty Through the Use of Interval Type-2 Modeling and General Type-2 Fuzzy Logic Agent Based Methods*. PhD thesis, University of Michigan, 2011.

[32] Gray, P., K. B. and Mavris, D., "Parametric Heat Exchanger Design for Next Generation Advanced Vehicle and Propulsion System Analysis," in *48th AIAA-ASME-SAE-ASEE Joint Propulsion Conference and Exhibit*, (Atlanta, GA), American Institute of Aeronautics and Astronautics, July 2012.

[33] Guynn, M., Berton, J., Tong, M., and Haller, W., "Advanced Single-Aisle Transport Propulsion Design Options Revisited," in *2013 Aviation Technology, Integration and Operations Conference*, pp. 1–17, 2013.

[34] Harry, D., "Optimization in Solid Rocket Booster Application," in *Proceedings of the 4th AIAA/NASA/USAF/OAI Symposium on Multidisciplinary Analysis and Optimization*, (Cleveland, OH), AIAA Paper No. 92-4688, September 1992.

[35] Hastie, T., T. R. and Friedman, J., *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer, 2 ed., 2009.

[36] Hopp, W. and Spearman, M., *Factory Physics*. McGraw-Hill Irwin, 3rd ed., 2008.

[37] James, B. B., " Multidisciplinary Optimization of a Controlled Space Structure Using 150 Design Variables," in *Proceedings of the 4th AIAA/NASA/ USAF/OAI Symposium on Multidisciplinary Analysis and Optimization*, (Cleveland, OH), AIAA Paper No. 92-4754, September 1992.

[38] Jimenez, H., B. G. S. J. and Mavris, D. N., "Probabilistic Technology Assessment for NASA Environmentally Responsible Aviation (ERA) Vehicle Concepts," in *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, (Virginia Beach, VA), September 2011.

[39] Jimenez, H., P. H. and Mavris, D. N., "System-wide Fleet Assessment of NASA Environmentally Responsible Aviation (ERA) Technologies and Concepts for Fuel Burn and CO2," in *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, (Virginia Beach, VA), September 2011.

[40] Jimenez, H., P. H. H. B. R. and Mavris, D. N., "Noise Assessment of NASA Environmentally Responsible Aviation (ERA) Technologies and Concepts," in *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, (Indianapolis, IN), September 2012.

[41] JMP, *Design of Experiments Guide*. 2010.

[42] Kennedy, G., Kenway, G., and Martins, J., "High Aspect Ratio Wing Design: Optimal Aerostructural Tradeoffs for the Next Generation of Materials," in *52nd Aerospaces Sciences Meeting*, no. January, (National Harbor, MD), pp. 1–24, 2014.

[43] Kestner, B. K., S. J. S. G. J. C. and Mavris, D. N., "Ultra High Bypass Ratio Engine Sizing and Cycle Selection Study for a Subsonic Commercial Aircraft in the N+2 Timeframe," in *ASME Turbo Expo*, (Vancouver, Canada), 2011.

[44] Kestner, B. K., S. J. S. T. J. C. M. P. C. A., "Surrogate Modeling for Simultaneous Engine Cycle and Technology for Next Generation Subsonic Aircraft," in *ASME Turbo Expo*, (Copenhagen, Denmark), 2012.

[45] Kinney, J. and Gurinder, S., "Equitability, mutual information, and the maximal information coefficient," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 111, no. 9, pp. 3354–3359, 2014.

[46] Kirby, M., B. P. and Mavris, D., "Enhancing the Environmnetal Policy Making Process with the FAAs EDS Analysis Tool," in *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, (Reston, VA), American Institute of Aeronautics and Astro- nautics, January 2009.

[47] Kirby, M. R. and Mavris, D. N., "The Environmental Design Space," in *26th International Congress of the Aeronautical Sciences*, 2008.

[48] Kizer, J. R. and Mavris, D. N., "Set-based design space exploration enabled by dynamic constraint analysis," International Council of the Aeronautical Sciences, 2014.

[49] Kotsiantis, S., "Supervised machine learning: A review of classification techniques," *Informatica (lithuanian Academy of Sciences)*, vol. 31, pp. 249–268, 2007.

[50] Kraskov, A., S. H. and Grassberger, P., "Estimating mutual information," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 69, no. 6 Pt 2, 2004.

[51] Kraskov, A., S. H. A. R. and Grassberger, P., "Hierarchical clustering using mutual information," *EPL (Europhysics Letters)*, vol. 70, no. 2, pp. 278–284, 2005.

[52] Kybic, J., "High-dimensional Mutual Information Estimation for Image Registration," in *Image Processing, 2004 International Conference on Image Processing (ICIP)*, (Singapore), October 2004.

[53] Lavelle, T. and Plencner, R., "Concurrent Optimization of Airframe and Engine Design," in *Proceedings of the 4th AIAA/NASA/USAF/OAI Symposium on Multidisciplinary Analysis and Optimization*, (Cleveland, OH), AIAA Paper No. 92-4713, September 1992.

[54] Liker, J., *The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer*. McGraw-Hill, 1st ed., 2004.

[55] Loeppky, J. L., S. J. and Welch, W., "Choosing the sample size of a computer experiment: A practical guide.," *Technometrics*, vol. 51, no. 4, pp. 366–376, 2009.

[56] Loftin, L. K., *Quest for Performance: The Evolution of Modern Aircraft*. National Aeronautics and Space Administration, 1985.

[57] Malak, R. J., Aughenbaugh, J. M., and Paredis, C. J., "Multi-attribute utility analysis in set-based conceptual design," *Computer-Aided Design*, vol. 41, pp. 214–227, Mar. 2009.

[58] Mankins, J., "Technology readiness levels," *White Paper, April*, pp. 4–8, 1995.

[59] Mankins, J. C., "Technology readiness and risk assessments: A new approach," *Acta Astronautica*, vol. 65, pp. 1208–1215, Nov. 2009.

[60] Mankins, J. C., "Technology readiness assessments: A retrospective," *Acta Astronautica*, vol. 65, pp. 1216–1223, Nov. 2009.

[61] Marsh, G., "Boeing's 787: trials, tribulations, and restoring the dream," *Reinforced Plastics*, vol. 53, pp. 16–21, Nov. 2009.

[62] Mattingly, J. D., Heiser, W. H., and Pratt, D. T., *Aircraft Engine Design*. American Institute of Aeronautics and Astronautics, Inc., 2nd ed., 2002.

[63] Mavris, D. and DeLaurentis, D., "A stochastic approach to multi-disciplinary aircraft analysis and design," *36th Aerospace Sciences ...*, 1998.

[64] Missoum, S., Ramu, P., and Haftka, R., "A convex hull approach for the reliability-based design optimization of nonlinear transient dynamic problems," *Computer methods in applied mechanics ...*, vol. 196, pp. 2895–2906, 2007.

[65] Molino, N. A., S. J. D. S. J. O. E. and Mavris, D. N., "Improved Pareto Optimal Engine Cycle Designs Through the Use of a New Pareto Quality Indicator," in *48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, (Atlanta, GA), 2012.

[66] Monakhova, Y.B., A. S. K. A. M. S., "Independent components in spectroscopic analysis of complex mixtures," *Chemometrics and Intelligent Laboratory Systems*, vol. 103, no. 2, pp. 108–115, 2010.

[67] Morris, S. J. and Kroo, I., "Aircraft design optimization with dynamic performance constraints," *Journal of Aircraft*, vol. 27, no. 12, pp. 1060–1067, 1990.

[68] Myers, R. H., Montgomery, D. C., and Anderson-Cook, C. M., *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. John Wiley & Sons, Inc., 3rd ed., 2009.

[69] NIST, "A glossary of doe terminology." Online.

[70] Norris, G., "Ecodemonstrator paves technology pathway to future," October 2012.

[71] Norris, G., "Boeing's 787 ecodemonstrator goes to work," August 2014.

[72] Padula, S. L. and Kincaid, R. K., "Aerospace Applications of Integer and Combinatorial Optimization," in *NASA TM-110210*, October 1995.

[73] Perullo, C. A., H. B. R. and Tai, J. C. M., "An Integrated Assessment of an Advanced Open Rotor Configuration Using the Environmental Design Space," in *48th AIAA-ASME-SAE-ASEE Joint Propulsion Conference and Exhibit*, (Atlanta, GA), August 2012.

[74] Perullo, C. A., T. J. C. M. and Mavris, D. N., "Effects of Advanced Engine Technology on Open Rotor Cycle Selection and Performance," in *ASME Turbo Expo*, (Copenhagen, Denmark), 2012.

[75] Pfaender, H., "Environmental Challenge : How to Close the Gap Between Policy and Technology?," in *9th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, (Hilton Head, SC), September 2009.

[76] Pfaender, H. and Mavris, D. N., "Effect of Fuel Price on Aviation Technology and Environmental Outcomes," in *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, (Indianapolis, IN), September 2012.

[77] Pfaender, H., J. H. and Mavris, D. N., "Fleet Assessment of Fuel Burn and NOx Emissions for NASA Environmentally Responsible Aviation (ERA) Technologies and Concepts," in *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, (Indianapolis, IN), September 2012.

[78] Phillips, W. F., *Mechanics of Flight*. John Wiley & Sons, Inc., 1st ed., 2004.

[79] Preparata, F. P. and Shamos, M. I., *Computational Geometry: An Introduction*. Springer-Verlag, 1985.

[80] Prescott, P., "Orthogonal-column Latin hypercube designs with small samples," *Computational Statistics & Data Analysis*, vol. 53, pp. 1191–1200, Feb. 2009.

[81] Price, H., "Fast sheet-continuous lower energy, emissions, and noise - cleen -program." Online.

[82] Pronzato, L. and Müller, W. G., "Design of computer experiments: space filling and beyond," *Statistics and Computing*, vol. 22, pp. 681–701, Apr. 2011.

[83] R Core Team, *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013.

[84] Raymer, D. P., *Aircraft Design: A Conceptual Approach*. American Institute of Aeronautics and Astronautics, Inc., 4th ed., 2006.

[85] Reddy, E., A. G. M. P. and Chamis, C., "Structural Tailoring of Aircraft Engine Blade Subject to Ice Impact Constraints," in *Proceedings of the 4th AIAA/NASA/USAF/OAI Symposium on Multidisciplinary Analysis and Optimization*, (Cleveland, OH), AIAA Paper No. 92-4710, September 1992.

[86] Sacks, J., Welch, W., Mitchell, T., and Wynn, H., "Design and analysis of computer experiments," *Statistical science*, 1989.

[87] Sands, J. S., G. J. C. K. B. K. and Mavris, D. N., "Hybrid Wing Body Engine Cycle Design Exploration for Boundary Layer Ingesting (BLI) Propulsion Systems Under Design Uncertainty," in *48th AIAA-ASME-SAE-ASEE Joint Propulsion Conference and Exhibit*, (Atlanta, GA), August 2012.

[88] Schutte, J. S., J. H. and Mavris, D. N., "Technology Assessment of NASA Environmentally Responsible Aviation Advanced Vehicle Concepts," in *49th AIAA Aerospace Sciences Meeting*, (Orlando, FL), January 2011.

[89] Schutte, J. S., K. B. T. J. and Mavris, D. N., "Updates and Modeling Enhancements to the Assessment of NASA Environmentally Responsible Aviation Technologies and Vehicle Concepts," in *50th AIAA Aerospace Sciences Meeting*, (Nashville, TN), 2012.

[90] Simpson, T., Booker, A., and Ghosh, D., "Approximation methods in multidisciplinary analysis and optimization: a panel discussion," in *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis & Optimization*, (Atlanta), pp. 1–16, 2004.

[91] Simpson, T., Lin, D., and Chen, W., "Sampling strategies for computer experiments: design and analysis," *International Journal of Reliability and Applications*, 2001.

[92] Sobek, D., Ward, A., and Liker, J., "Toyota's principles of set-based concurrent engineering," in *Sloan management review*, pp. 67–83, 1999.

[93] Sobieszczanski-Sobieski, J. and Haftka, R., "Multidisciplinary aerospace design optimization: Survey of recent developments," *Structural Optimization*, vol. 14, no. 1, pp. 1–23, 1997.

[94] Stgbauer, H., A. R. K. A. and Grassberger, P., "Reliability of ICA estimates with mutual information," in *Independent Component Analysis and Blind Signal Separation Lecture Notes in Computer Science*, pp. 209–216, 2004.

[95] Stgbauer, H., K. A. A. S. and Grassberger, P., "Least-dependent-component analysis based on mutual information," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, no. 6 Pt 2, p. 70, 2004.

[96] Stoll, R. R., *Sets, Logic and Axiomatic Theories*. W.H. Freeman and Company, 2nd ed., 1974.

[97] Swiler, L., Slepoy, R., and Giunta, A., "Evaluation of Sampling Methods in Constructing Response Surface Approximations," *47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference¡BR¿ 14th AIAA/ASME/AHS Adaptive Structures Conference¡BR¿ 7th*, pp. 1–24, May 2006.

[98] Taguchi, G., *System of Experimental Design : Engineering Methods to Optimize Quality and Minimize Costs*. White Plains, N.Y. : Dearborn, Mich.: UNIPUB/Kraus International Publications ; American Supplier Institute, 6 ed., 1987.

[99] Thain, D., Tannenbaum, T., and Livny, M., "Distributed computing in practice: the condor experience.," *Concurrency - Practice and Experience*, vol. 17, no. 2-4, pp. 323–356, 2005.

[100] Tzong, G., B. M. D. R. and Giesing, J., "Aeroelastic Loads and Structural Optimization of a High Speed Civil Transport Model," in *Proceedings of the 5th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, (Panama City Beach, FL), AIAA Paper 94-4378, September 1994.

[101] Vanderplaats, G. N., *Multidiscipline Design Optimization*. Vanderplaats Research & Development, Inc., 1st ed., 2007.

[102] Viana, F. a. C., Venter, G., and Balabanov, V., "An algorithm for fast optimal Latin hypercube design of experiments," *International Journal for Numerical Methods in Engineering*, vol. 2, pp. 135–156, 2009.

[103] Ward, A., Liker, J., and Sobek, D. I., "The second toyota paradox: How delaying decisions can make better cars faster," *Sloan Managment Review*, vol. 36, no. 2, pp. 43–61, 1995.

[104] Washburn, A., "An overview of nasa's environmentally responsible aviation project plan: 2013 - 2015," tech. rep., National Aeronautics and Space Administration, January 2013.

[105] Weisstein, E. W., "Hypercube." Online.

[106] Wu, X., K. V. Q. J. R. G. J. Y. Q. M. H. M. G. N. A. L. B. Y. P. Z. Z. S. M. H. D. and Steinberg, D., "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol. 14, pp. 1–37, 2008.

[107] Zentner, J. M., *A Design Space Exploration Process for Large Scale , Multi-objective Computer Simulations*. PhD thesis, Georgia Institute of Technology, 2006.

VITA

Justin R. Kizer was born and raised in Houston, Texas. He attended undergraduate school at the University of Texas at Austin where he received a B.S. in Aerospace Engineering in May 2010. While attending UT Austin, Justin was active within both the Design Build Fly and AUVSI student clubs, participating in the design and construction of Unmanned Aerial Systems. Justin also conducted research regarding the vapor plumes of Enceladus as a student researcher within the Aeromechanics Research Group. Seeking to expand his knowledge of systems engineering and the design of advanced concepts, he attended graduate school at the Georgia Institute of Technology. As a graduate researcher within the Aerospace Systems Design Laboratory, Justin conducted research regarding advanced concept design and technology forecasting with several sponsors including the Office of Naval Research and NASA. Justin earned his M.S. in Aerospace Engineering from Georgia Tech in December 2011 and ultimately graduated with a Ph.D. in Aerospace Engineering in 2016.